

# **Compact / CANopen / Logic Controller / Twido System User Guide**



# Contents

<b>Important Information .....</b>	<b>3</b>
<b>Before You Begin.....</b>	<b>4</b>
<b>Introduction .....</b>	<b>6</b>
<b>Glossary .....</b>	<b>8</b>
<b>Application Source Code .....</b>	<b>9</b>
<b>Typical Applications .....</b>	<b>10</b>
<b>System .....</b>	<b>12</b>
<b>Architecture.....</b>	<b>12</b>
<b>Installation.....</b>	<b>15</b>
<b>Hardware .....</b>	<b>18</b>
<b>Software .....</b>	<b>26</b>
<b>Communication .....</b>	<b>27</b>
<b>Implementation .....</b>	<b>34</b>
<b>Communication .....</b>	<b>35</b>
<b>Controller .....</b>	<b>38</b>
<b>HMI.....</b>	<b>67</b>
<b>Devices .....</b>	<b>91</b>
<b>Altivar 312 .....</b>	<b>92</b>
<b>TeSysU .....</b>	<b>104</b>
<b>Appendix.....</b>	<b>106</b>
<b>Detailed Component List.....</b>	<b>106</b>
<b>Component Protection Classes.....</b>	<b>108</b>
<b>Component Features .....</b>	<b>109</b>
<b>Contact.....</b>	<b>113</b>

## Important Information

**NOTICE** Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

### **DANGER**

**DANGER** indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

### **WARNING**

**WARNING** indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

### **CAUTION**

**CAUTION** indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

### **CAUTION**

**CAUTION**, used without the safety alert symbol, indicates a potentially hazardous situation which, if not avoided, **can result in** equipment damage.


**PLEASE NOTE** Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and the installation, and has received safety training to recognize and avoid the hazards involved

© 2008 Schneider Electric. All Rights Reserved.

## Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

 <b>WARNING</b>
<b>UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY</b> <ul style="list-style-type: none"><li>• Do not use this software and related automation products on equipment which does not have point-of-operation protection.</li><li>• Do not reach into machine during operation.</li></ul> <b>Failure to follow these instructions can cause death, serious injury or equipment damage.</b>

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation and maintenance of the machine; therefore, only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to the applicable local and national standards and regulations. A "National Safety Council's" Accident Prevention Manual also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products by itself cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks for point-of-operation protection have been installed and are operational before placing the equipment into service. All mechanical/electrical interlocks and safeties for point-of-operation protection must be coordinated with the related automation equipment and software programming.

**NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of this document.**

### START UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.



## CAUTION

### EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters and debris from equipment.

**Failure to follow these instructions can result in injury or equipment damage.**

Follow all start up tests recommended in the equipment documentation. Store all equipment documentation for future reference.

**Software testing must be done in both simulated and real environments.**

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove ground from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

## OPERATION AND ADJUSTMENTS

The following precautions are from NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and rating of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

## WARNING

### UNINTENDED EQUIPMENT OPERATION

- Only use software tools approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

**Failure to follow these instructions can cause death, serious injury or equipment damage.**

# Introduction

---

## Introduction

This document is intended to provide a quick introduction to the described system. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers additional information to the product documentation, for installing, configuring and implementing the system.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider Electric and third-party components may be integrated to fulfill an industrial application.

A detailed functional description or the specification for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The architecture described in this document has been fully tested in our laboratories using all the specific references you will find in the component list near the end of this document. Of course, your specific application requirements may be different and will require additional and/or different components. In this case, you will have to adapt the information provided in this document to your particular needs. To do so, you will need to consult the specific product documentation of the components that you are substituting in this architecture. Pay particular attention in conforming to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

It should be noted that there are some major components in the architecture described in this document that cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified herein. You must be aware of the consequences of component substitution in the architecture described in this document as substitutions may impair the compatibility and interoperability of software and hardware.

<b>⚠ CAUTION</b>
<b>EQUIPMENT INCOMPATIBILITY OR INOPERABLE EQUIPMENT</b>
Read and thoroughly understand all hardware and software documentation before attempting any component substitutions.
<b>Failure to follow these instructions can result in injury or equipment damage.</b>

## Abbreviations

Abbreviation	Signification
<b>AC</b>	Alternating Current
<b>CB</b>	Circuit Breaker
<b>DI</b>	Digital Input
<b>DO</b>	Digital Output
<b>DC</b>	Direct Current
<b>EDS</b>	Electronic Data Sheet
<b>E-STOP</b>	Emergency Stop
<b>HMI</b>	Human Machine Interface
<b>I/O</b>	Input/Output
<b>IL</b>	Instruction List - a textual IEC-61131 programming language
<b>LD</b>	Ladder Diagram - a graphic IEC-61131 programming language
<b>PC</b>	Personal Computer
<b>PDO</b>	Process Data Object (CANopen)
<b>PS</b>	Power Supply
<b>RPM</b>	Revolutions Per Minute
<b>RPDO</b>	Receive Process Data Object (CANopen)
<b>SE</b>	Schneider Electric
<b>SDO</b>	Service Data Object
<b>TPDO</b>	Transmit Process Data Object (CANopen)
<b>TVDA</b>	Tested, Validated, Documented Architectures
<b>VSD</b>	Variable Speed Drive
<b>WxHxD</b>	Dimensions: Width, Height and Depth

## Glossary

---

Expression	Signification
<b>Altivar (ATV)</b>	SE product name for a family of VSDs
<b>CANopen</b>	Name for a communications machine bus system
<b>Harmony</b>	SE product name for a family of switches and indicators
<b>Magelis</b>	SE product name for a family of HMI devices
<b>MB – SL</b>	SE name for a serial Modbus communications protocol
<b>Modbus</b>	A communications protocol
<b>OsiSense</b>	SE product name for a family of sensors
<b>Phaseo</b>	SE product name for a family of power supplies
<b>SoMove</b>	SE product name for a drive configuration software
<b>Preventa</b>	SE product name for a family of safety devices
<b>TeSys</b>	SE product name for a family for motor protection devices and load contactors
<b>Twido</b>	SE product name of a basic range family of controllers
<b>TwidoSuite</b>	SE product name for a controller programming software
<b>Vijeo Designer</b>	SE product name for Magelis HMI devices configuration software

---

# Application Source Code

## Introduction

The example source code is in the form of configuration, application and import files. Use the appropriate software tool to either open or import the files.

Extension	File Type	Software Tool Required
<b>CFG</b>	Export File (Multiloader Export)	SoMove (Lite)
<b>DOC</b>	Document file	Microsoft Word
<b>EDS</b>	<b>E</b> lectronic <b>D</b> ata <b>S</b> heet - Device Definition	Industrial standard
<b>PDF</b>	Portable Document Format - document	Adobe Acrobat
<b>PSX</b>	Project file	SoMove (Lite)
<b>SPA</b>	Schneider Product Archive	TwidoSuite
<b>VDZ</b>	Project file	Vijeo Designer
<b>XPR</b>	Project file	TwidoSuite
<b>ZW1</b>	Project file	EPLAN P8

# Typical Applications

---

## Introduction

Here you will find a list of the typical applications, and their market segments, where this system or subsystem can be applied:

### Packaging

- Filling machines
- Bottling machines
- Wrapping machines
- Corking machines
- Pallet wrappers

### Textile

- Clothing machines
- Sawing machines

### Pumping

- Booster stations
- Compressors
- Vacuum pumps
- Hydraulic, Air, Filter pumps...

### HVAC-R

- Compressors

### Other Machines

- Wood working machines
- Cutting machines
- Sanders
- Sawing machines
- Industrial washing machines

### **SPECIAL NOTE**

The products specified in this document have been tested under actual service conditions. Of course, your specific application requirements may be different from those assumed for this and any related examples described herein. In that case, you will have to adapt the information provided in this and other related documents to your particular needs. To do so, you will need to consult the specific product documentation of the hardware and/or software components that you may add or substitute for any examples specified in this documentation. Pay particular attention and conform to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

The application examples and descriptions contained in this document have been developed based on products and standards available and defined for Europe. Some or all of the application examples may contain recommendations of products that are not available in your country or locality, or may recommend wiring, products, procedures or functions that are in conflict with your local, regional or national electrical or safety codes and/or normative standards.

### **▲ WARNING**

#### **REGULATORY INCOMPATIBILITY**

Be sure that all equipment applied and systems designed comply with all applicable local, regional and national regulations and standards.

**Failure to follow these instructions can result in death, serious injury, or equipment damage.**

### **NOTE:**

The information in this document is based on European standards and may not be valid for use in the U.S.A.

The use and application of the information contained herein require expertise in the design and programming of automated control systems. Only the user or integrator can be aware of all the conditions and factors present during installation and setup, operation, and maintenance of the machine or process, and can therefore determine the automation and associated equipment and the related safety provisions and interlocks which can be effectively and properly used. When selecting automation and control equipment, and any other related equipment or software, for a particular application, the user or integrator must also consider any applicable local, regional or national standards and/or regulations.

---

# System

## Introduction

The system chapter describes the architecture, dimensions, quantities, required software and the different types of components used within this system.

## Architecture

### General

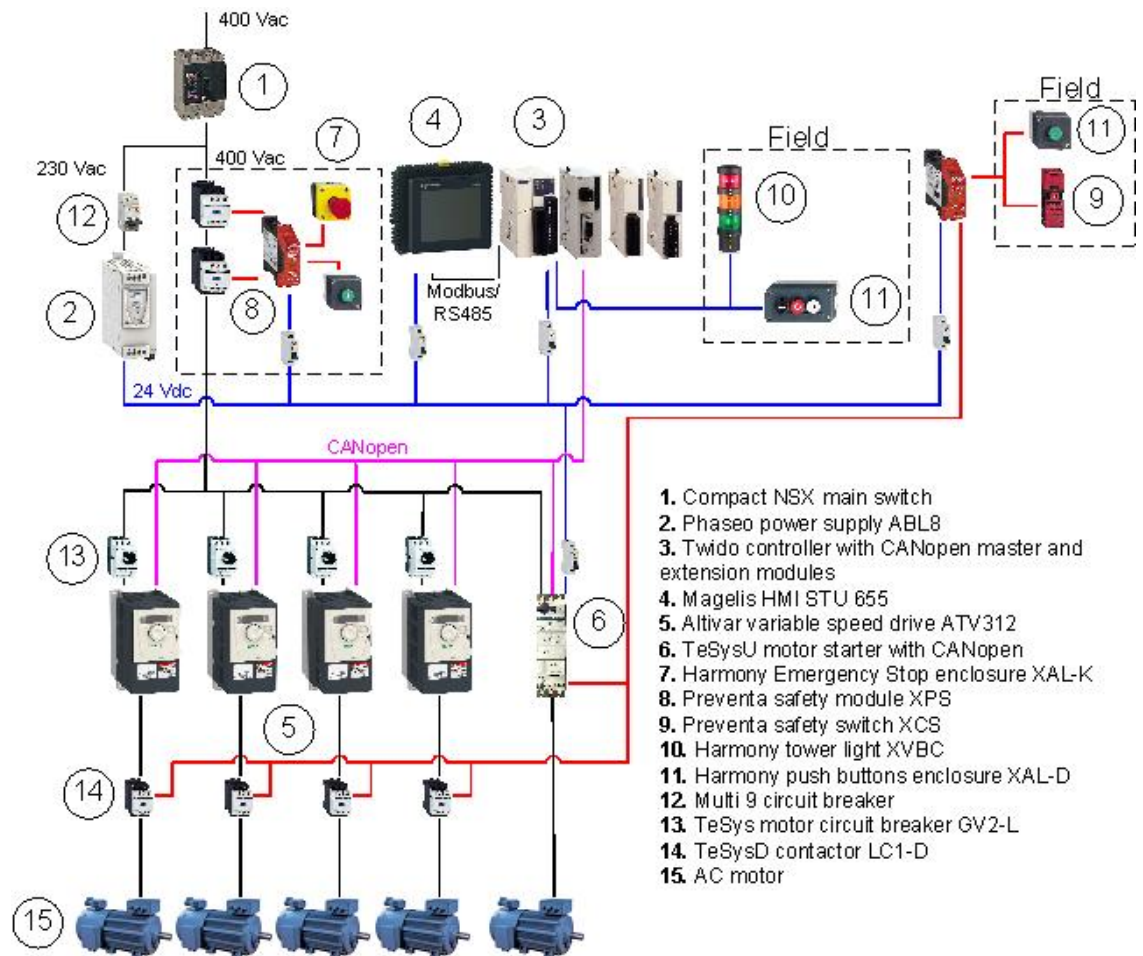
The controller in this application is a Twido. The user can control the application using the Magelis HMI.

The variable speed drives, connected to the Twido via CANopen bus, are of the type Altivar 312. An additional motor is driven by a TeSysU motor starter.

The example application includes two functional safety options:

1. A tamper free Emergency Stop function supervised with a Preventa safety module.
2. A second safety module to evaluate protective door guard function.

### Layout





<b>Components</b>	<b>Hardware:</b>			<ul style="list-style-type: none"> <li>• Compact main switch NSX100F</li> <li>• Motor circuit breaker GV2L</li> <li>• TeSysD contactor LC1D</li> <li>• Altivar variable speed drive ATV312</li> <li>• TeSysU motor starter</li> <li>• Harmony XALK Emergency Stop switch with rotation release</li> <li>• Phaseo power supply ABL8</li> <li>• Modicon Twido modular controller</li> <li>• CANopen tap TSX</li> <li>• Magelis HMISTU655 Graphic display terminal</li> <li>• Harmony XB5 selector switches, push buttons and indicator lamps</li> <li>• Harmony tower light</li> <li>• Preventa guard switch with mechanical actuator XCS</li> <li>• Preventa safety module XPS</li> <li>• Multi 9 circuit breaker</li> <li>• Standard AC motor</li> </ul>
	<b>Software:</b>			<ul style="list-style-type: none"> <li>• Twidosuite Version 2.20.11</li> <li>• SoMove 1.2.4.0</li> <li>• Vijeo Designer 5.1 SP1</li> </ul>
<b>Quantities of Components</b>	For a complete and detailed list of components, the quantities required and the order numbers, please refer to the detailed components list in the appendix.			
<b>Degree of Protection</b>	Not all the components in this configuration are designed to withstand the same environmental conditions. Some components may need additional protection, in the form of housings, depending on the environment in which you intend to use them. For environmental details of the individual components please refer to the list in the appendix of this document and the appropriate user manual.			
<b>Cabinet Technical Data</b>	<b>Input</b>	Mains voltage	400 Vac	
		Power requirement	~ 3 kW	
		Cable Size	5 x 2.5 mm <sup>2</sup> (L1, L2, L3, N, PE)	
		Cable Connection	3 phase + Neutral + Ground	
			Neutral is needed for 230 Vac (Phase and Neutral)	
	<b>Output</b>	Motor power ratings	4 asynchronous motors (4 poles: 1500 RPM) controlled by ATV312 (0.37 kW)	
			1 asynchronous motors (4 poles: 1500 RPM) controlled by TeSysU (0.18 kW)	

---

**Functional  
Safety Notice****(EN ISO13849-1  
EN IEC62061)**

The standard and level of functional safety you apply to your application is determined by your system design and the overall extent to which your system may be a hazard to people and machinery.

Whether or not a specific functional safety category should be applied to your system should be ascertained with a proper risk analysis.

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the functional safety requirements with respect to the equipment used in their systems or of compliance with either national or international safety laws and regulations

**Emergency  
Stop****Emergency Stop/Emergency Disconnection function**

This function for stopping in an emergency is a protective measure which complements the safety functions for the safeguarding of hazardous zones according to EN ISO 12100-2.

**Safety Function****Door guarding:**

**up to Performance Level (PL) = b, Safety Integrity Level (SIL) = 1**

---

**Dimensions**

The compact dimensions of the devices used, for example, the controller and power supply, enable the components to be installed inside a small control panel with the following external dimensions: 800 x 1200 x 400 mm (WxHxD).

The display elements used to indicate "Plant Ready" and "Safety Acknowledged" can be built into the door of the control cabinet along with the system master switch and Emergency Stop switch.

---

# Installation

## Introduction

This chapter describes the steps necessary to set up the hardware and configure the software required to fulfill the described function of the application.

## Assembly



## Notes

The components and I/O points listed below represent a cross-section of the components and signals which can be used in conjunction with most typical applications (functional safety/maintenance switches) and are a basic necessity for control and display purposes and a number of optional inputs and outputs.

The components designed for installation in a control cabinet, example Twido, Phaseo power supply unit, Harmony Emergency Stop switching device, line circuit breaker, contactors and motor circuit breaker, etc. can be snapped onto a 35 mm DIN rail.

The Altivar variable speed drive is installed directly on the mounting plate.

Emergency Stop, main switch and HMI are installed in the cabinet door.

The Preventa guard switch is mounted in the field.

There are two options available for mounting Harmony XB5 push buttons and indicator lamps:

- a) Using a 22 mm hole drilled into the front door of the control cabinet in the appropriate position.
- b) Using XALD housing, this can house up to 5 push buttons or indicator lamps. This XALD is designed for backplane assembly or direct wall mounting.

The individual components must be interconnected in accordance with the detailed circuit diagram to help ensure that they function correctly.

- 400 Vac wiring between Compact main switch, motor circuit breaker, load relay and variable speed drives.
  - 24 Vdc wiring between power supply unit, Twido, push buttons, indicator lamps and variable speed drive control circuit.
  - 24 Vdc between Emergency Stop, Preventa, and the control circuit of the load contactor.
-

## Twido I/O Wiring

Twido inputs	Description
I 0.0	Not used, reserved for fast counters
I 0.1	Not used, reserved for fast counters
I 0.2	Safety Relay 1, Emergency Stop
I 0.3	Safety Relay 2, door guarding
I 0.4	Motor protection switch, drive 1
I 0.5	Motor protection switch, drive 2
I 0.6	Motor protection switch, drive 3
I 0.7	Motor protection switch, drive 4
I 0.8	Push button start
I 0.9	Push button stop
I 0.10	Not used
I 0.11	Not used

Twido outputs	Description
Trans. Out Q0	Pilot light, acknowledge button, Emergency stop
Trans. Out Q1	Pilot light, acknowledge button, door guarding
Relay Out Q2	Pilot light, start button
Relay Out Q3	Pilot light, stop button
Relay Out Q4	Pilot light, plant ready for operation
Relay Out Q5	Not used
Relay Out Q6	Not used
Relay Out Q7	Not used

## Digital extension module

Digital I/O-Module inputs	Description
all inputs	Not used

Digital I/O-Module outputs	Description
Relay Out Q0	Tower light, green indicator lamp
Relay Out Q1	Tower light, red indicator lamp
Relay Out Q2	Tower light, blue indicator lamp
Relay Out Q3	Tower light, white indicator lamp
Relay Out Q4	Not used
Relay Out Q5	Not used
Relay Out Q6	Not used
Relay Out Q7	Not used

## Analog extension module

Analog I/Os	Description
all inputs and outputs	Not used


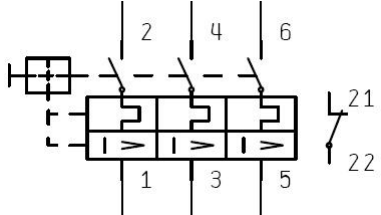

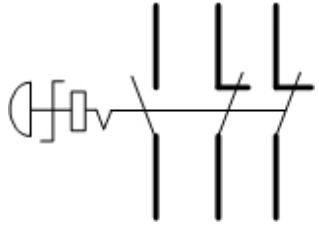

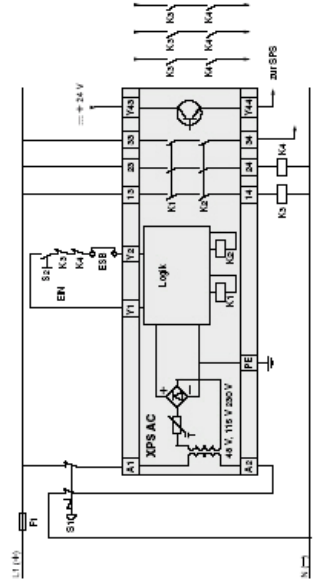
## Thermo- couple module


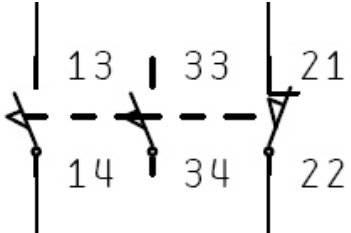

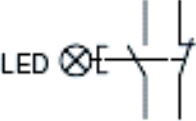

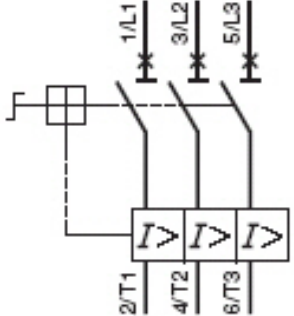

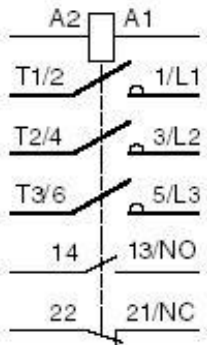
Connections	Description
Out0	Not used
In0	Temperature sensor Pt100
In1	Temperature sensor Pt100


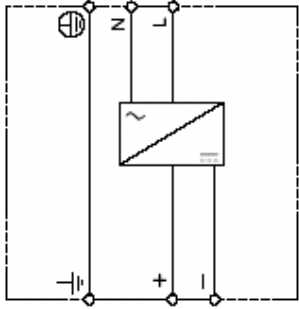

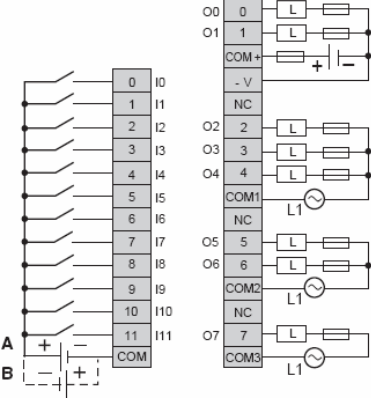

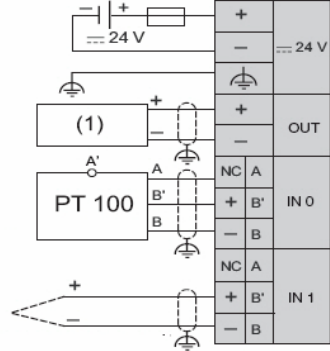
Twido CANopen module	Description
COM	CANopen bus communication

Twido 24 Vdc supply	Description
COM	Supply inputs 0 Vdc
COM+	Supply outputs +24 Vdc
-V	Supply outputs 0 Vdc
COM 1	Supply outputs +24 Vdc
COM 2	Supply outputs +24 Vdc
COM 3	Supply outputs +24 Vdc

# Hardware

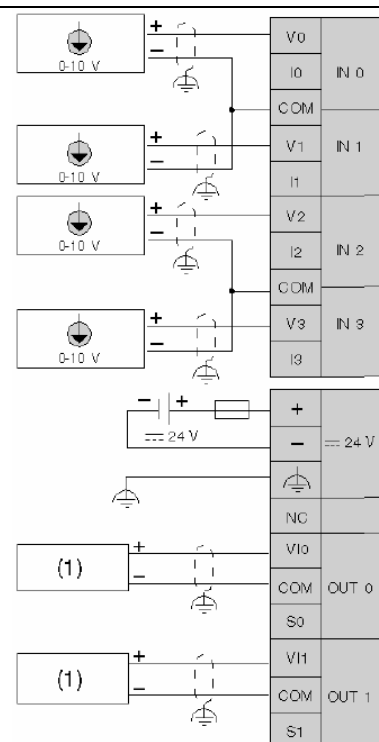
<b>Main Switch</b>  <b>Compact</b>  <b>NSX100F</b>		
<b>Emergency Stop for door mounting</b>  <b>Harmony</b>  <b>XB5AS844 + XB5AZ141</b>  Incl. Emergency Stop Label <b>ZBY8330</b>		
<b>Emergency Stop Safety Module</b>  <b>Preventa</b>  <b>XPSAC5121</b>		

<p><b>Guard switch</b></p> <p><b>Preventa</b></p> <p><b>XCSA502</b></p>		
<p><b>Indicator Press Button</b></p> <p><b>Harmony</b></p> <p><b>XB5</b></p>		
<p><b>Motor circuit breaker</b></p> <p><b>TeSys</b></p> <p><b>GV2L07</b></p>		
<p><b>Contactor</b></p> <p><b>TeSysD</b></p> <p><b>LC1D09BD</b></p>		

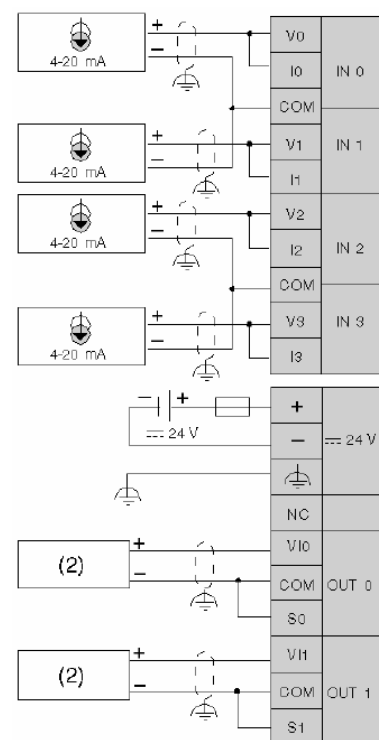
<p><b>Power supply</b></p> <p><b>Phaseo</b></p> <p><b>ABL8RPS24050</b></p> <p>24 Vdc, 5 A</p>		
<p><b>Twido</b></p> <p><b>Logic controller</b></p> <p>Modular device</p> <p><b>TWDLMDA20DRT</b></p> <p>12 digital inputs 8 relay outputs</p>		 <p><b>A</b> = Input wiring positive logic <b>B</b> = Input wiring negative logic</p>
<p><b>Twido</b></p> <p><b>Analog expansion module</b></p> <p><b>TM2ALM3LT</b></p> <p>2 analog inputs 1 analog output</p> <p>12 bit</p>		 <p>(1) Voltage/current preactuator</p> <p>Module covers thermocouple (types J, K and T) and temperatureprobe (Pt100)</p> <p><b>Note:</b> For 4 wires A' is not connected</p>



4 analog inputs  
2 analog output


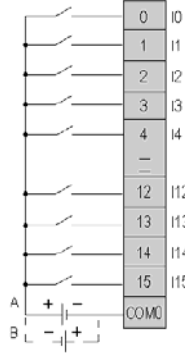
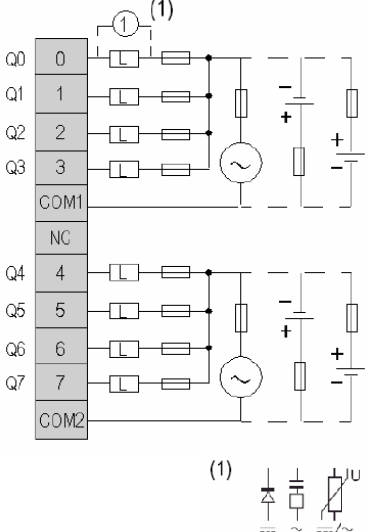

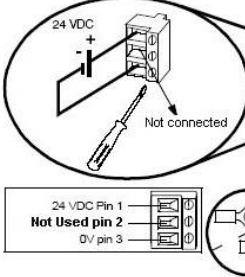


(1) Voltage preactuator



(2) Current preactuator

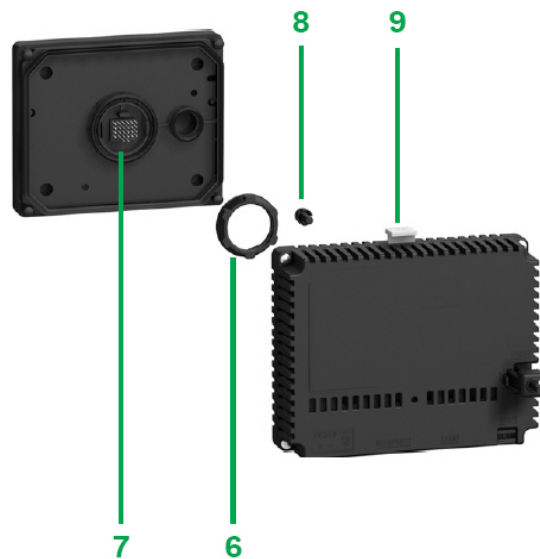
Use the braid supplied with the module to connect the functional ground

<p><b>Twido</b></p> <p><b>Digital expansion module</b></p> <p><b>TM2DMM24DRF</b></p> <p>16 digital inputs 8 digital outputs</p>		<p><b>Inputs</b></p>  <p><b>A</b> = Input wiring positive logic <b>B</b> = Input wiring negative logic</p> <p><b>Outputs</b></p> 																														
<p><b>Twido</b></p> <p><b>Interface Module CANopen Master</b></p> <p><b>TWDNCO1M</b></p>		<p><b>CAN Power Supply</b></p>  <p><b>CAN Bus Cable Connections</b></p> <table border="1"> <thead> <tr> <th>Pin</th> <th>Signal</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Not Used</td> <td>Reserved</td> </tr> <tr> <td>2</td> <td>CAN_L</td> <td>CAN-L-Bus Cable</td> </tr> <tr> <td>3</td> <td>CAN_GND</td> <td>CAN-Earth</td> </tr> <tr> <td>4</td> <td>Not Used</td> <td>Reserved</td> </tr> <tr> <td>5</td> <td>CAN_SHLD</td> <td>Optional CAN-Shield</td> </tr> <tr> <td>6</td> <td>GND</td> <td>Optional Earth</td> </tr> <tr> <td>7</td> <td>CAN_H</td> <td>CAN-H-Bus Cable</td> </tr> <tr> <td>8</td> <td>Not Used</td> <td>Reserved</td> </tr> <tr> <td>9</td> <td>CAN_V+</td> <td>Not Connected</td> </tr> </tbody> </table> <p>AWG 12 (2,5 mm<sup>2</sup>) &lt; 2 m</p>	Pin	Signal	Description	1	Not Used	Reserved	2	CAN_L	CAN-L-Bus Cable	3	CAN_GND	CAN-Earth	4	Not Used	Reserved	5	CAN_SHLD	Optional CAN-Shield	6	GND	Optional Earth	7	CAN_H	CAN-H-Bus Cable	8	Not Used	Reserved	9	CAN_V+	Not Connected
Pin	Signal	Description																														
1	Not Used	Reserved																														
2	CAN_L	CAN-L-Bus Cable																														
3	CAN_GND	CAN-Earth																														
4	Not Used	Reserved																														
5	CAN_SHLD	Optional CAN-Shield																														
6	GND	Optional Earth																														
7	CAN_H	CAN-H-Bus Cable																														
8	Not Used	Reserved																														
9	CAN_V+	Not Connected																														

**Operator panel**  
**Magelis HMI**  
**HMISTU655**


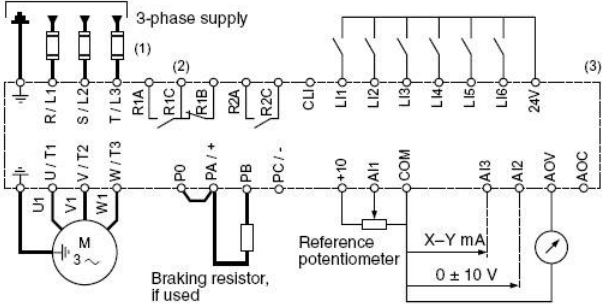

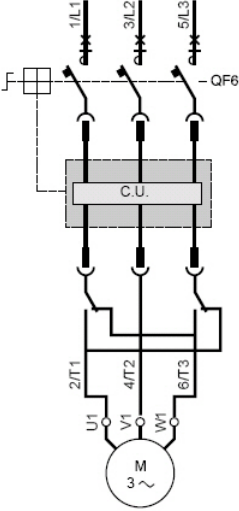



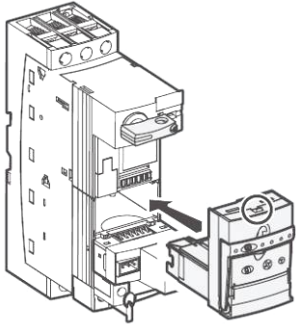

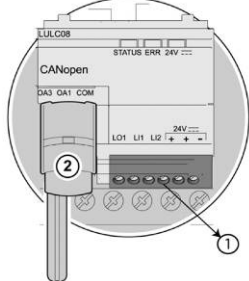
*Exploded view of Magelis STU Small Panel: simple installation by means of a 22 mm diameter hole*



A Magelis STU Small Panel is made up of a front module (comprising the screen) and a rear module (comprising the CPU plus terminals and connectors). The two modules are fixed together by means of a hole measuring 22mm in diameter. The fixing system contains the following elements:

- (6) An adjusting nut
- (7) A seal
- (8) An anti rotation tee (can be used as an option)
- (9) A release mechanism: simply press to separate the two modules once they have been fixed together

<p><b>Variable Speed Drive</b></p> <p><b>Altivar 312</b></p> <p><b>ATV312H037N4</b></p> <p>3-phase, 400 Vac, 0.37 kW</p>		
		
<p><b>Motor Starter</b></p> <p><b>TeSysU</b></p> <p>Power base two directions</p> <p><b>LUB12BL</b></p> <p>Coil wiring kit</p> <p><b>LULC08</b></p>		

<p><b>Motor Starter</b></p> <p><b>TeSysU</b></p> <p>Control Unit</p> <p><b>LUCA05BL</b></p>	 <p>The image shows a TeSysU LUCA05BL control unit. It is a white, rectangular device with a blue Telemecanique logo at the top. Below the logo, it says '24V ~'. The main body has a large circular dial with current ratings: 1.25A, 1.70A, 2.10A, 3.00A, 3.80A, and 5.00A. Below the dial, it says 'CLASS 10 3~'.</p>	 <p>The diagram shows a motor starter assembly with a terminal block on the right. A wire is connected from the terminal block to the control unit. The terminal block has terminals for L1, L2, L3, and N.</p>
<p><b>Motor Starter</b></p> <p><b>TeSysU</b></p> <p>CANopen communication module</p> <p><b>LULC08</b></p>	 <p>The image shows a LULC08 CANopen communication module. It is a white, rectangular device with a blue Telemecanique logo at the top. Below the logo, it says 'LULC08'. The main body has a large circular dial with current ratings: 1.25A, 1.70A, 2.10A, 3.00A, 3.80A, and 5.00A. Below the dial, it says 'CLASS 10 3~'.</p>	 <p>The diagram shows the LULC08 module with a terminal block. A wire is connected from the terminal block to the module. The terminal block has terminals for L1, L2, L3, and N. A legend indicates: 1. 24 Vdc power Supply, 2. Terminal for coil wiring kit.</p> <ol style="list-style-type: none"> <li>1. 24 Vdc power Supply</li> <li>2. Terminal for coil wiring kit</li> </ol>

# Software

---

## General

**TwidoSuite** is used for programming the Twido, including the configuration for communication and assigning inputs and outputs.

The HMI application on the Magelis operator terminal is configured using Vijeo Designer software.

The Altivar 312 variable speed drive can be configured using the keypad panel. However, the SoMove software is a more user-friendly option and can be used for parameterizing the drive, saving data and quickly restoring existing data for service purposes. The software can also help you to optimize the parameters online.

To use the software packages, your PC must have the appropriate Microsoft Windows operating system installed:

- Windows XP Professional

The default installation path on the hard drive of your PC for each of the software tools is:

### TwidoSuite

C:\Program Files\Schneider Electric\TwidoSuite



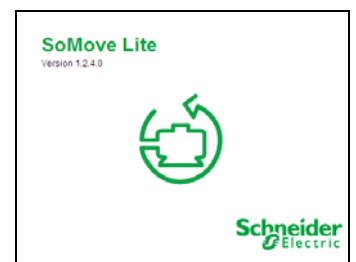
### Vijeo Designer

C:\Program Files\Schneider Electric\Vijeo-Designer



### SoMove Lite

C:\Program Files\Schneider Electric\SoMove Lite



# Communication



## General

A Modbus connection is used to exchange data between the Magelis HMI terminal and the Twido. The XBTZ9780 communication cable shown below is required to connect these two devices. The software driver required for Modbus communication is already included in the software packages for the Magelis panel and the Twido.

A CANopen link is used to connect the controller, the Altivar 312 and the TeSysU device. The different cables, adapters and connectors are explained on the following pages to allow the wiring inside the cabinet.

This chapter describes the hardware used for data communications (CANopen) and how to use the programming cables used in configuring the individual devices in the system.

## HMI

<b>HMI &lt;&gt; PC</b>	HMISTU terminals can be connected to the PC in two different ways: <ul style="list-style-type: none"><li>- via USB port</li><li>- via Ethernet</li></ul>
<b>Download cable PC &lt;&gt; HMI</b>  <b>BMXXCAUSBH018</b>	 <i>BMX XCA USB0●●</i>
<b>Communication Cable HMI &lt;&gt; controller</b>  <b>XBTZ9780</b>	 XBT Z9780 (RS 485) RJ45 MiniDin

Controller

**Communication cable with USB connector**

**PC <=> Twido**

**TSXCRJMD25**

To program the Twido, use the cable **TSXCRJMD25** with the RS485 adapters (**TSXCUSB485**) to connect the PC with the Twido.

Verify that the adapter is in switch position **2**.

For the communication to work properly, port 1 must be configured on the Twido

TSX CUSB485  
USB to RS485 converter  
Quick Reference Guide

TSX CUSB485  
Konverter USB-RS485  
Kurzanleitung

TSX CUSB485  
Convertisseur USB-RS485  
Instruction de service

TSX CUSB485  
Convertidor USB-RS485  
Guia de referencias rápidas

TSX CUSB485  
Convertitore USB-RS485  
Guida di riferimento

TSX CUSB485  
USB 至 RS485 转换器  
快速参考指南

OTB 1S0DM9LP

PLC :  
Micro  
Premium  
Twido

XBTN/R/S

Unitelway  
Tap  
TSXSCA62

RJ45

Minidin8

RJ45

DB15

VW3 A8306R..

TSX C RJMD 25

with power

XBTZ925

TSX CRJDB 25

RJ45

USB-A

Windows 2000, XP

USB-A

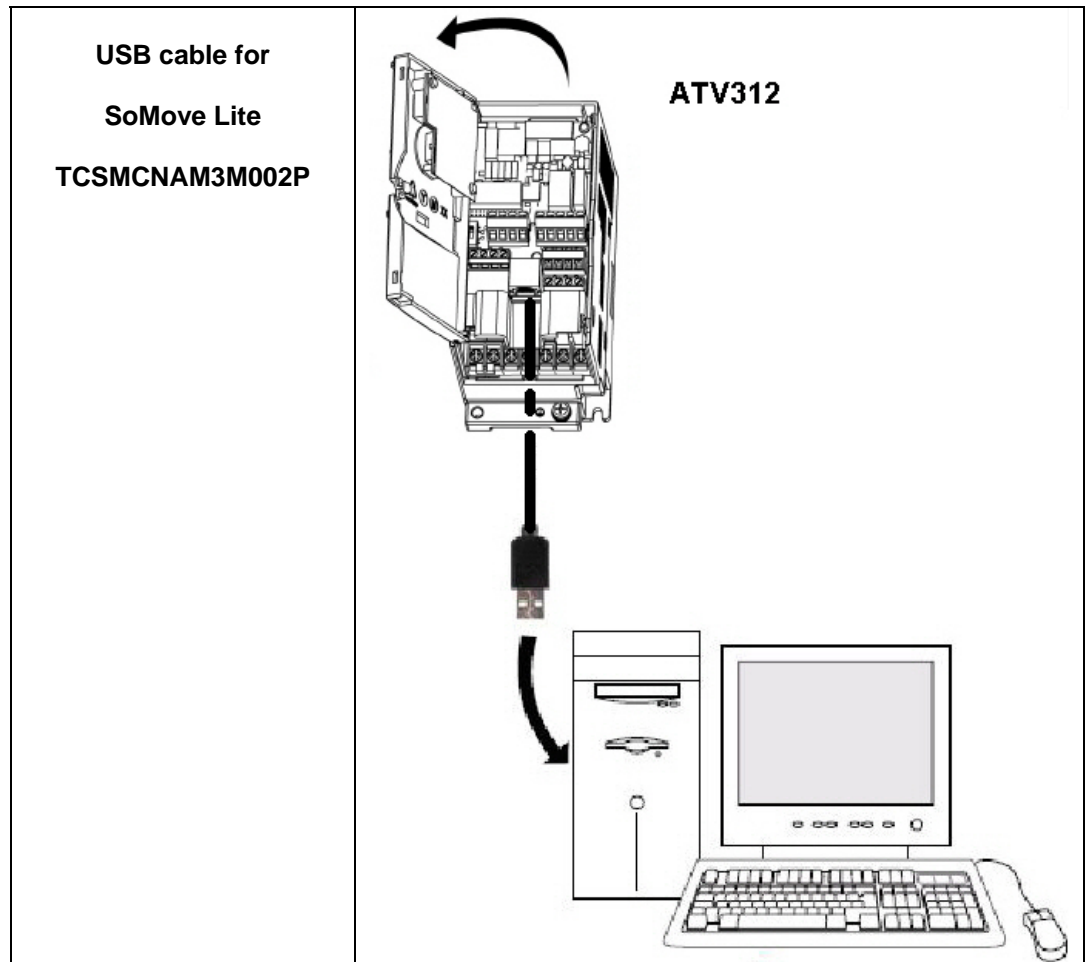
1	VBUS (+5V)
2	Data (D-)
3	Data (D+)
4	U-GND

RJ45

1	N.C.	
2	N.C.	
3	PMC	Port Mode Control
4	D1 (B)	Transceiver Terminal 1
5	D0 (A)	Transceiver Terminal 0
6	NDE	RS485 Driver Enable (INPUT)
7	VP (OUT)	Power Supply (5V, 500 mA max)
8	Common	Signal and power supply Common



## Altivar 312

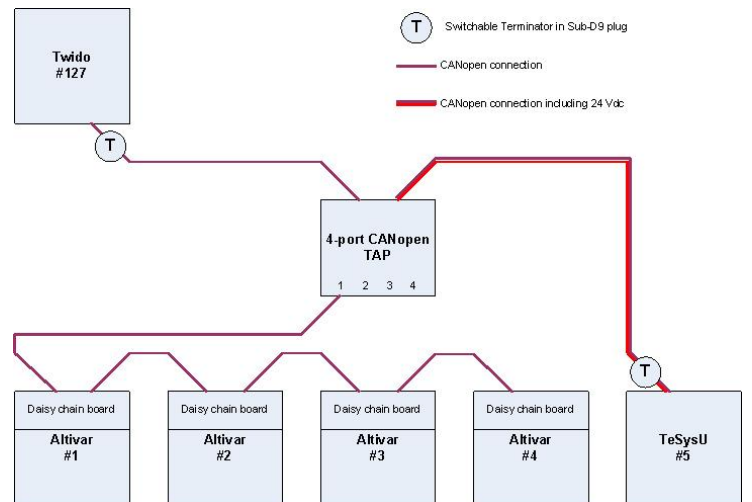




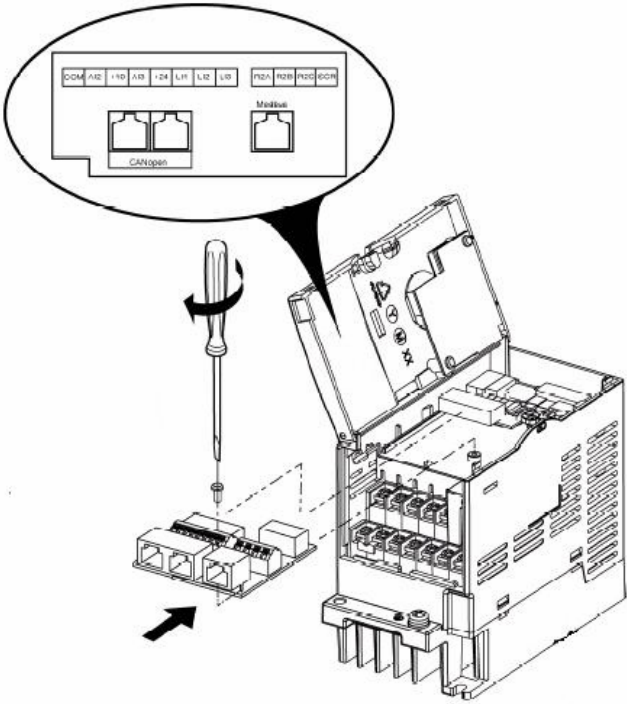
## CANopen

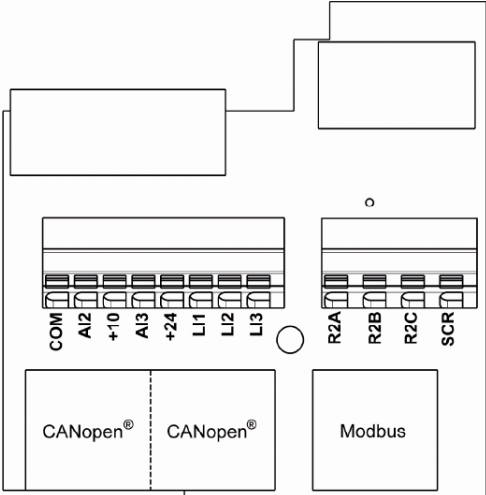
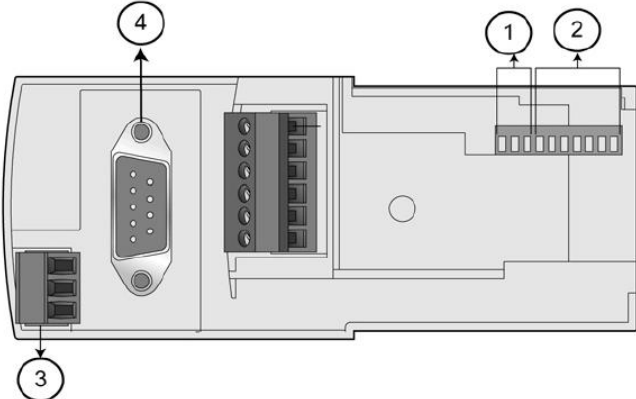
### CANopen topology used in this architecture

The picture shows where the CANopen bus needs to be terminated.

Precondition for the terminating resistors is that the entire cable length of the stub (connected to connection 1 on the CANopen TAP) is shorter than the cables between Twido and CANopen TAP and TeSysU and CANopen TAP.



<p><b>CANopen Master Module</b></p> <p><b>TWDNCO1M</b></p> <p>The CANopen Interface Module is plugged into the Twido and has a SUB-D9 CANopen connection</p>	
<p><b>Altivar 312</b></p> <p><b>CANopen</b></p> <p>daisy chain option</p> <p><b>Node ID: 1, 2, 3 and 4</b></p>	
<p><b>ATV312</b></p> <p><b>CANopen option card</b></p> <p>daisy chain option</p> <p><b>VW3A31208</b></p>	

<p><b>ATV312</b></p> <p><b>CANopen</b></p> <p>daisy chain option card</p> <p><b>VW3A31208</b></p>																																					
<p><b>TeSysU CANopen communication module</b></p> <p><b>LULC08</b></p> <p><b>Node ID: 5</b></p>	 <div><p>1 Baud rate</p><p>2 Address</p><p>3 Power base connector</p><p>4 CAN bus connector</p></div>																																				
<p><b>TeSysU CANopen communication module</b></p> <p><b>LULC08</b></p> <p>The baudrate is set to</p> <p><b>500 kbps.</b></p>	<table><tr><th>SW10</th><th>SW9</th><th>SW8</th><th>Baud Rate</th></tr><tr><td>0</td><td>0</td><td>0</td><td>10 kbps</td></tr><tr><td>0</td><td>0</td><td>1</td><td>20 kbps</td></tr><tr><td>0</td><td>1</td><td>0</td><td>50 kbps</td></tr><tr><td>0</td><td>1</td><td>1</td><td>125 kbps</td></tr><tr><td>1</td><td>0</td><td>0</td><td>250 kbps (default)</td></tr><tr><td>1</td><td>0</td><td>1</td><td>500 kbps</td></tr><tr><td>1</td><td>1</td><td>0</td><td>800 kbps</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1,000 kbps</td></tr></table>	SW10	SW9	SW8	Baud Rate	0	0	0	10 kbps	0	0	1	20 kbps	0	1	0	50 kbps	0	1	1	125 kbps	1	0	0	250 kbps (default)	1	0	1	500 kbps	1	1	0	800 kbps	1	1	1	1,000 kbps
SW10	SW9	SW8	Baud Rate																																		
0	0	0	10 kbps																																		
0	0	1	20 kbps																																		
0	1	0	50 kbps																																		
0	1	1	125 kbps																																		
1	0	0	250 kbps (default)																																		
1	0	1	500 kbps																																		
1	1	0	800 kbps																																		
1	1	1	1,000 kbps																																		

## TeSysU CANopen communication module

The following address is used: **Node ID: 5**

SW7	SW6	SW5	SW4	SW3	SW2	SW1	Address
0	0	0	0	0	0	0	Not valid
0	0	0	0	0	0	1	1 (default)
0	0	0	0	0	1	0	2
0	0	0	0	0	1	1	3
0	0	0	0	1	0	0	4
0	0	0	0	1	0	1	5

### CANopen Connector

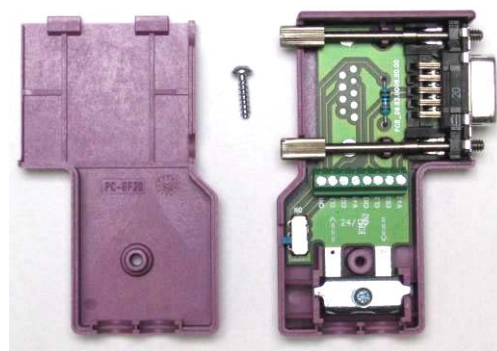
#### TSXCANKCDF90T

Use this connector on the **TWDNCO1M**. Connector includes a terminal resistor.

CANopen cable

#### TSXCANCD50

Flexible Cable



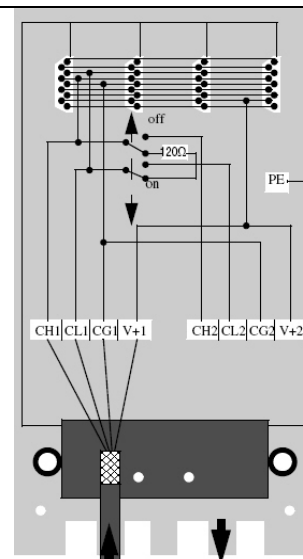
### CANopen TAP

#### TSXCANTDM4

Terminal resistor can be selected using the **ON /OFF** switch.

The image shows the resistor set to **OFF**.

As TeSysU bus module requires a 24 Vdc power supply, this needs to be wired on the CANopen out of the CANopen TAP.



### CANopen cable

#### TCSCCN4F3M1T

Connects from the CANopen TAP to the first Altivar 312.



**CANopen  
pre-assembled  
connection cable**

**VW3CANCARR03  
(length: 0,3m)**



Used for connecting ATV312 in a daisy chain connection.

# Implementation

## Introduction

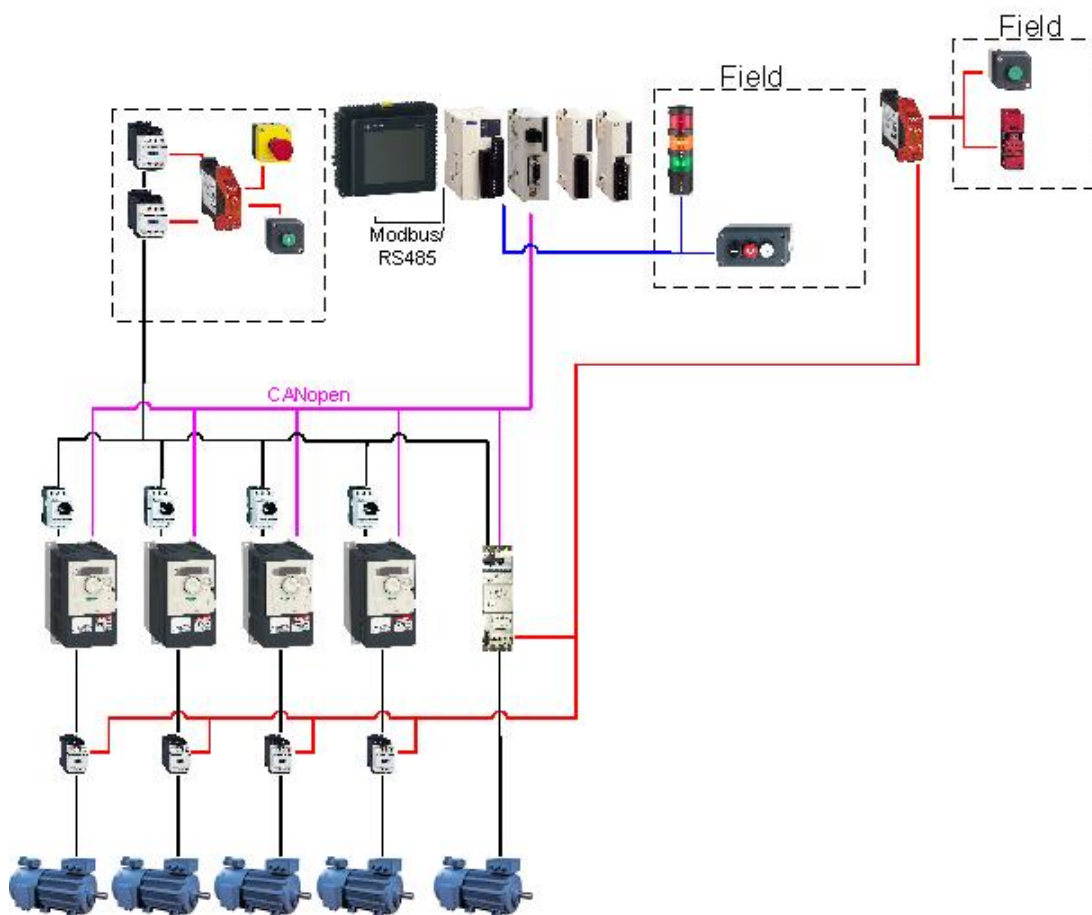
The implementation chapter describes all the steps necessary to initialize, configure, program and start-up the system to achieve the application functions as listed below.

## Function

### Start-up and functional description

1. Switch on all circuit breakers. The main switch is off.
2. Verify that the Emergency Stop has not been activated
3. Switch on at the main switch
4. Check the door guard switch
5. Acknowledge the Emergency Stop and door guard switch
6. The start-up screen is shown on the HMI touch screen
7. Press the **Bus** button on the right to open the CANopen status page
8. Wait until all devices show a green status field and press the **Control Dev. 1-3** (panel for Altivar 1..3) or **Control Dev. 4&5** button (panel for Altivar 4 or TeSysU)
9. Select from the control screen per Altivar the setpoint speed, forward, reverse or stop.
10. Press the **Control Dev. 4&5** button (panel for Altivar 4 and TeSysU)
11. Select from the control screen per Altivar the setpoint speed, forward, reverse or stop. For TeSys U select only forward or revers.
12. In case an alarm occurs the alarm button in the right down corner will turn red. Press the alarm button to open the alarm summary page.

## Functional Layout



# Communication

## Introduction

This chapter describes the data passed via the communications networks (e.g. CANopen or Modbus) that is not bound directly with digital or analog hardware.

The list contains:

- The device links
- Direction of data flow
- Symbolic name and
- Bus address of the device concerned.

## Device Links

This application uses Modbus and CANopen networks.

**Modbus** connects:

- Magelis Panel as Modbus master
- Twido as Modbus slave (address 1)

**CANopen** connects the following devices:

- Twido CANopen master module (address 127)
- Altivar 312 variable speed drives (addresses 1..4)
- TeSysU reverse motor starter (address 5)

## Datalink HMI → Twido

Device 1		Device 2	
HMISTO511 (Modbus master)		Twido (Modbus slave 1)	
Address	Designation	Address	Name
%M11	HMI_Slave_1_FWD	%M11	HMI_SLAVE_1_FWD
%M12	HMI_Slave_1_REV	%M12	HMI_SLAVE_1_REV
%M13	HMI_Slave_1_STOP	%M13	HMI_SLAVE_1_STOP
%M21	HMI_Slave_2_FWD	%M21	HMI_SLAVE_2_FWD
%M22	HMI_Slave_2_REV	%M22	HMI_SLAVE_2_REV
%M23	HMI_Slave_2_STOP	%M23	HMI_SLAVE_2_STOP
%M31	HMI_Slave_3_FWD	%M31	HMI_SLAVE_3_FWD
%M32	HMI_Slave_3_REV	%M32	HMI_SLAVE_3_REV
%M33	HMI_Slave_3_STOP	%M33	HMI_SLAVE_3_STOP
%M41	HMI_Slave_4_FWD	%M41	HMI_SLAVE_4_FWD
%M42	HMI_Slave_4_REV	%M42	HMI_SLAVE_4_REV
%M43	HMI_Slave_4_STOP	%M43	HMI_SLAVE_4_STOP
%M51	HMI_Slave_5_FWD	%M51	HMI_SLAVE_5_FWD
%M52	HMI_Slave_5_REV	%M52	HMI_SLAVE_5_REV
%M53	HMI_Slave_5_STOP	%M53	HMI_SLAVE_5_STOP
%M100	HMI_Alarm_Reset	%M100	HMI_CLEAR_ERROR
%MW1	HMI_Panel_ID_Current	%MW1	HMI_PANEL_ID_CURRENT
%MW11	HMI_Slave_1_SP	%MW11	HMI_SETP_SLAVE_1
%MW12	HMI_Slave_2_SP	%MW12	HMI_SETP_SLAVE_2
%MW13	HMI_Slave_3_SP	%MW13	HMI_SETP_SLAVE_3
%MW14	HMI_Slave_4_SP	%MW14	HMI_SETP_SLAVE_4

**Datalink**  
**Twido → HMI**

Device 1		Device 2	
HMISTO511 (Modbus master)		Twido (Modbus slave 1)	
Address	Designation	Address	Name
%M2	HMI_global_Alarm_Stat	%M2	STATUS_ALARM_SUMMARY
%M101	HMI_Alarm_01	%M101	HMI_ALARM_01
%M102	HMI_Alarm_02	%M102	HMI_ALARM_02
%M103	HMI_Alarm_03	%M103	HMI_ALARM_03
%M104	HMI_Alarm_04	%M104	HMI_ALARM_04
%M105	HMI_Alarm_05	%M105	HMI_ALARM_05
%M106	HMI_Alarm_06	%M106	HMI_ALARM_06
%M107	HMI_Alarm_07	%M107	HMI_ALARM_07
%M108	HMI_Alarm_08	%M108	HMI_ALARM_08
%MW2	HMI_Panel_ID_Next	%MW2	HMI_PANEL_ID_NEXT
%MW11	HMI_Slave_1_SP	%MW11	HMI_SETP_SLAVE_1
%MW12	HMI_Slave_2_SP	%MW12	HMI_SETP_SLAVE_2
%MW13	HMI_Slave_3_SP	%MW13	HMI_SETP_SLAVE_3
%MW14	HMI_Slave_4_SP	%MW14	HMI_SETP_SLAVE_4
%MW20:X0	CANopen_Master_OK	%MW20:X0	CANOPEN_MASTER_OK
%MW20:X1	CANopen_Slave_1_OK	%MW20:X1	CANOPEN_SLAVE_1_OK
%MW20:X2	CANopen_Slave_2_OK	%MW20:X2	CANOPEN_SLAVE_2_OK
%MW20:X3	CANopen_Slave_3_OK	%MW20:X3	CANOPEN_SLAVE_3_OK
%MW20:X4	CANopen_Slave_4_OK	%MW20:X4	CANOPEN_SLAVE_4_OK
%MW20:X5	CANopen_Slave_5_OK	%MW20:X5	CANOPEN_SLAVE_5_OK
%MW31	HMI_Temp_1	%MD31	HMI_TEMPERATURE_1
%MW32	HMI_Temp_2	%MD32	HMI_TEMPERATURE_2

**Datalink**  
**Drive1 →**  
**Twido**

Drive 1	
Altivar (CANopen slave 1) to Twido (CANopen master)	
Address	Designation
%IWC1.0.0	D_STATUS_SLAVE_1
%IWC1.0.1	D_CONTROL_SLAVE_1
%IWC1.0.2	D_IERROR_SLAVE_1
%IWC1.0.3	D_MCURRENT_SLAVE_1
Twido (CANopen master) to Altivar (CANopen slave 1)	
Address	Designation
%QWC1.0.0	D_COMMAND_SLAVE_1
%QWC1.0.1	D_TARGET_SLAVE_1

**Twido →**  
**Drive1**

**Datalink**  
**Drive2 →**  
**Twido**

Drive 2	
Altivar (CANopen slave 2) to Twido (CANopen master)	
Address	Designation
%IWC1.1.0	D_STATUS_SLAVE_2
%IWC1.1.1	D_CONTROL_SLAVE_2
%IWC1.1.2	D_IERROR_SLAVE_2
%IWC1.1.3	D_MCURRENT_SLAVE_2
Twido (CANopen master) to Altivar (CANopen slave 2)	
Address	Designation
%QWC1.1.0	D_COMMAND_SLAVE_2
%QWC1.1.1	D_TARGET_SLAVE_2

**Twido →**  
**Drive2**



**Datalink  
Drive3 →  
Twido**

<b>Drive 3</b>	
<b>Altivar (CANopen slave 3) to Twido (CANopen master)</b>	
Address	Designation
%IWC1.2.0	D_STATUS_SLAVE_3
%IWC1.2.1	D_CONTROL_SLAVE_3
%IWC1.2.2	D_IERROR_SLAVE_3
%IWC1.2.3	D_MCURRENT_SLAVE_3
<b>Twido (CANopen master) to Altivar (CANopen slave 3)</b>	
Address	Designation
%QWC1.2.0	D_COMMAND_SLAVE_3
%QWC1.2.1	D_TARGET_SLAVE_3

**Twido →  
Drive3**

**Datalink  
Drive4 →  
Twido**

<b>Drive 4</b>	
<b>Altivar (CANopen slave 4) to Twido (CANopen master)</b>	
Address	Designation
%IWC1.3.0	D_STATUS_SLAVE_4
%IWC1.3.1	D_CONTROL_SLAVE_4
%IWC1.3.2	D_IERROR_SLAVE_4
%IWC1.3.3	D_MCURRENT_SLAVE_4
<b>Twido (CANopen master) to Altivar (CANopen slave 4)</b>	
Address	Designation
%QWC1.3.0	D_COMMAND_SLAVE_4
%QWC1.3.1	D_TARGET_SLAVE_4

**Twido →  
Drive4**

**Datalink  
TeSysU →  
Twido**

<b>TeSysU</b>	
<b>TeSysU (CANopen slave 5) to Twido (CANopen master)</b>	
Address	Designation
%IWC1.4.0	TPDO_1_2004SUB6 (internal TeSysU status register 455) *)
%IWC1.4.1	TPDO_1_2004SUB9 (internal TeSysU status register 458) *)
<b>Twido (CANopen Master) to Altivar (CANopen Slave 5)</b>	
Address	Designation
%QWC1.4.0	RPDO_1_2008SUB5 (internal TeSysU control register 704) *)
%QWC1.4.1	RPDO_2_2008SUB4 (internal TeSysU control register 703) *)
%QWC1.4.2	RPDO_3_2008SUB1 (internal TeSysU control register 700) *)

**Twido →  
TeSysU**

\*) only the first input word and output word are used in this example

# Controller

## Introduction

This chapter describes the steps required for the initialization and configuration and the source program required to fulfill the functions.

The controller program is created using TwidoSuite.

## Requirements

To use TwidoSuite and program the Twido first verify that:

- TwidoSuite is installed on your PC
- The example application, TwidoSuite project **Optimized\_CANopen\_Twido.xpr**, is in the standard project directory (C:\Program Files\Schneider Electric\TwidoSuite\My Projects)
- The Twido is switched on
- The Twido is connected to the PC using the proper programming cable

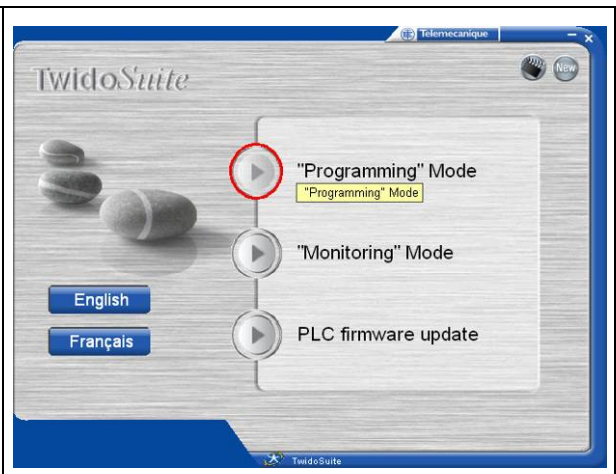
## Procedure

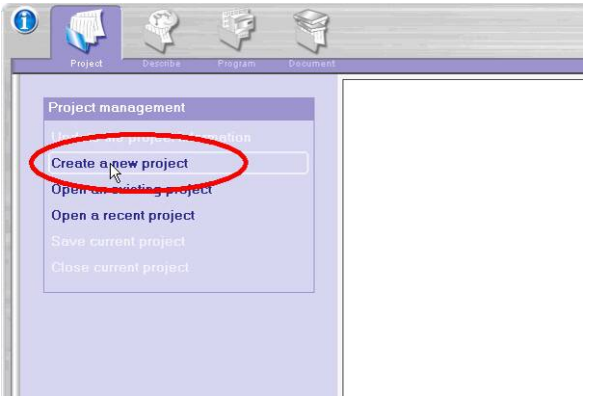
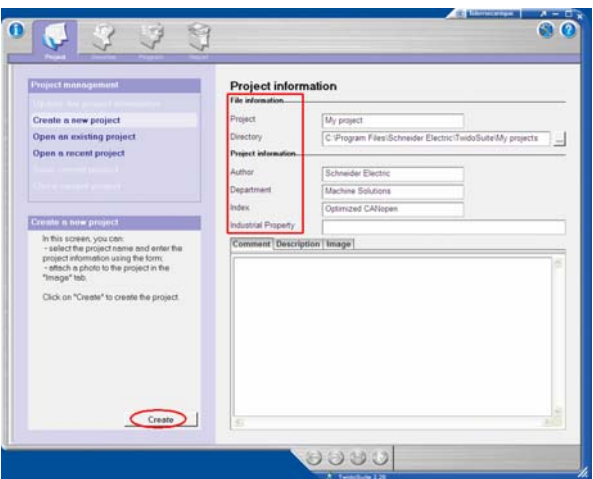
Programming the controller consists of:

- Create a new project
- Configure the Twido
- Configure the Modbus for HMI
- Import EDS-file for TeSysU
- Configure the CANopen devices
- CANopen module configuration
- Activate the Altivar drive macros
- Create variables
- Create an Application Program
- Insert Macros for ATV drives
- Control commands for TeSysU
- Save / export a project
- Open / import a project
- Connecting to a Twido
- Download a project
- Download firmware

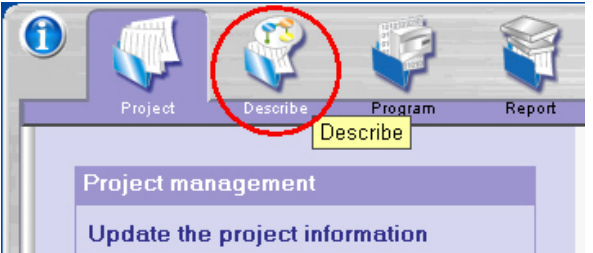
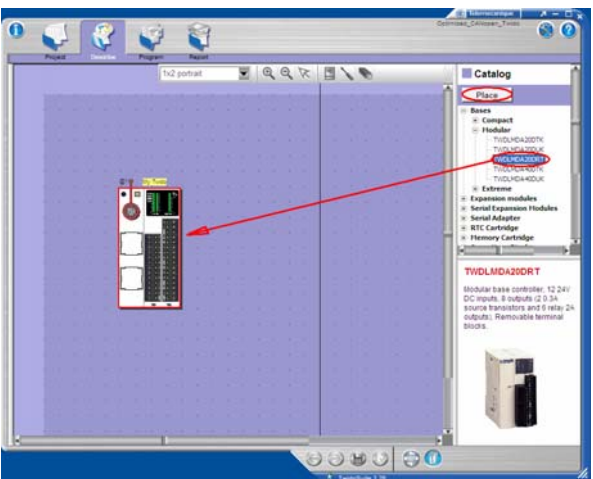
## Create a new project

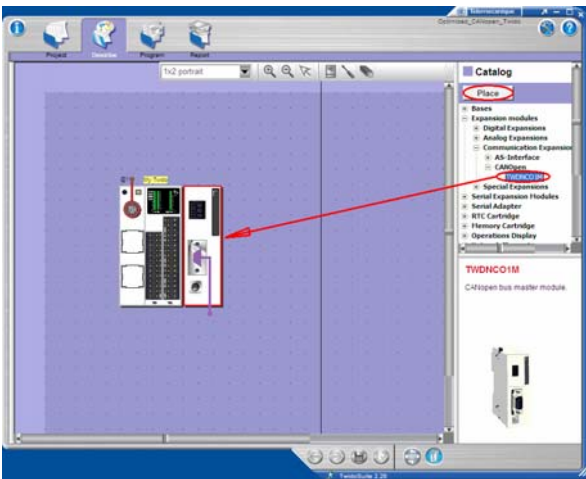
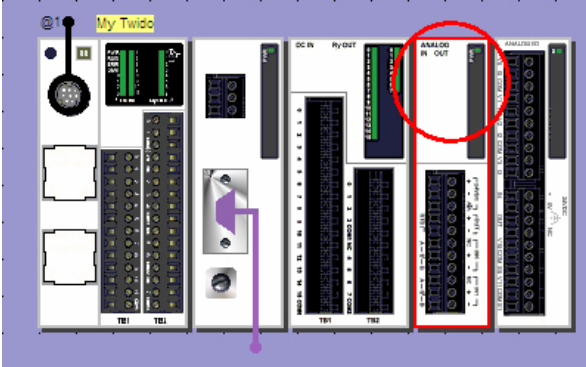
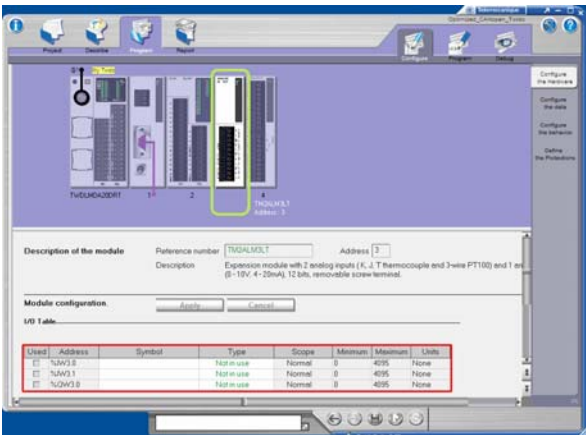
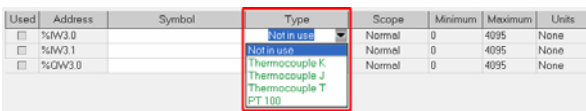
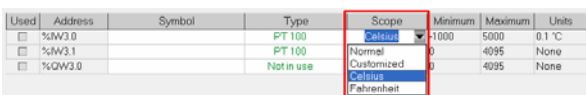
- |   |   |
|---|---|
| 1 | To create a new project click on the arrow next to<br><br><b>"Programming" Mode</b> |
|---|---|



2	<p>Select the <b>Project</b> tab and click on:</p> <p><b>Create a new project</b></p>	
3	<p>Enter the project information as required:</p> <ul style="list-style-type: none"> <li>• <b>Project name</b></li> <li>• <b>Directory path for save</b></li> <li>• <b>Author</b></li> <li>• <b>Department</b></li> <li>• <b>Index</b></li> <li>• <b>Industrial Property</b></li> </ul> <p>Continue with <b>Create</b>.</p>	

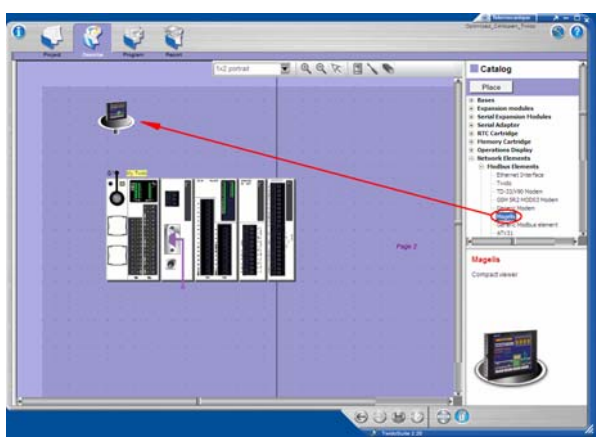
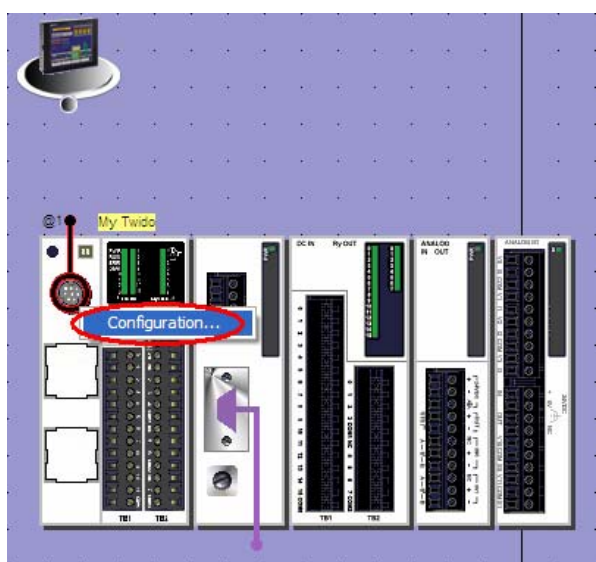
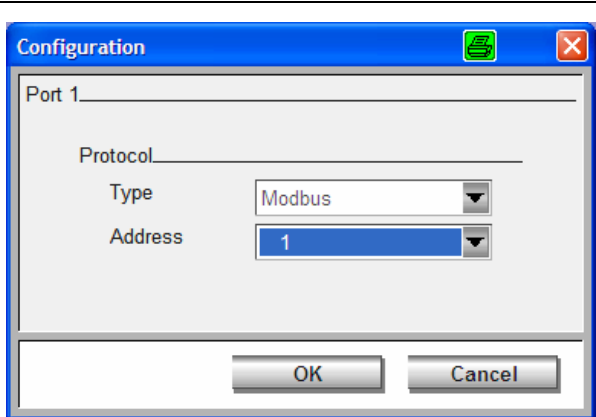
## Configure the Twido

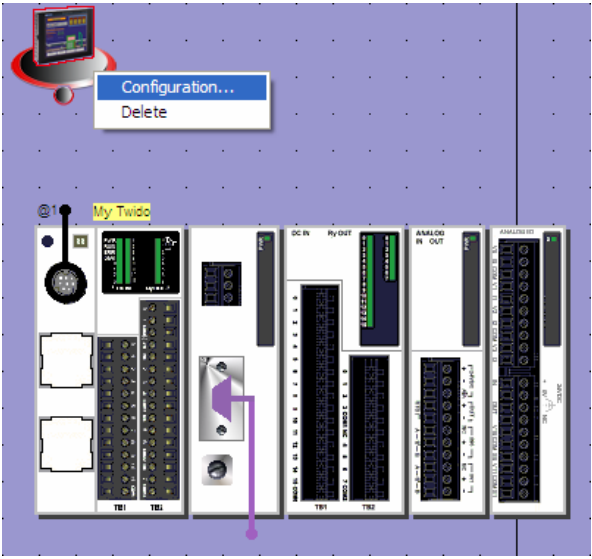
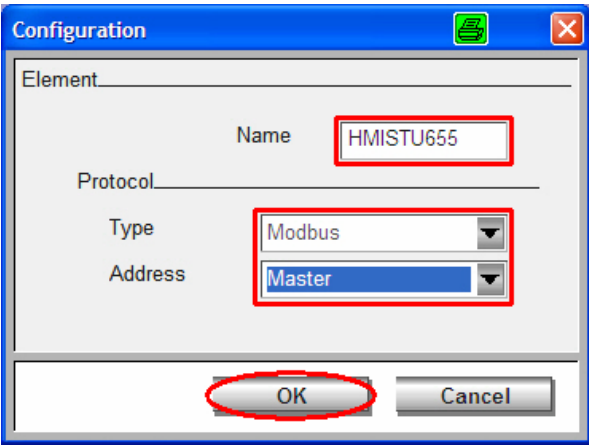
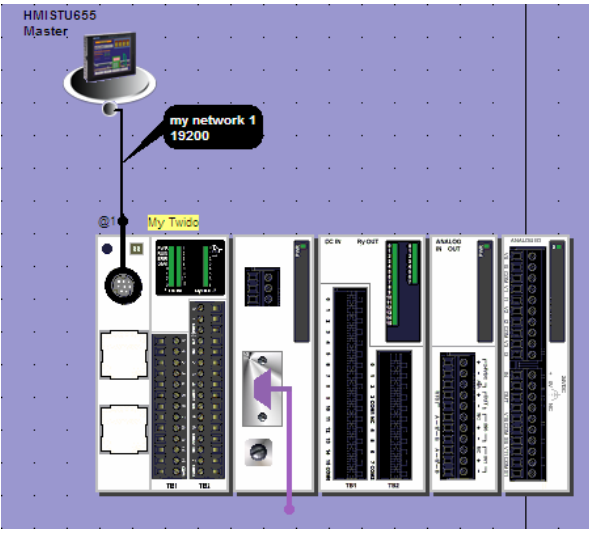
1	<p>To configure the hardware, the communication parameters and bus system, click on the</p> <p><b>Describe</b> tab.</p>	
2	<p>From the <b>Catalog</b> on the right select the Twido type:</p> <p><b>Bases-&gt;</b> <b>Modular-&gt;</b> <b>TWDLMDA20DRT</b></p> <p>Click on <b>Place</b> to see the Twido in the image of the rack.</p> <p>Pick the Twido from the list and drag it into the page. A green square indicates the possible position.</p>	

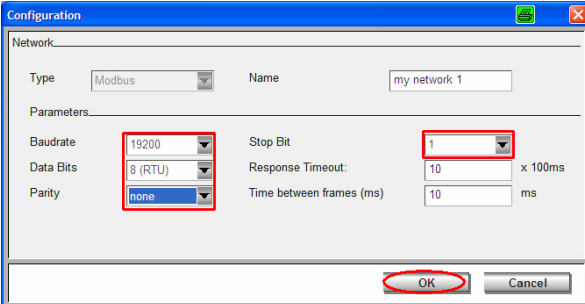
3	<p>Again in the <b>Catalog</b>, for the communications module, select:</p> <p><b>Expansions modules-&gt; Communication Expansion-&gt; CANopen-&gt;TWDNCO1M</b></p> <p>Click on <b>Place</b> and drag the item to the worktop to see the communications module show up in the image of the rack.</p> <p>Continue appending local I/Os like digital or analog inputs and outputs as described above.</p> <p>In this configuration following modules are used additionally:</p> <ul style="list-style-type: none"><li>• TM2DMM24DRF</li><li>• TM2ALM3LT</li><li>• TM2AMM6HT</li></ul>																																	
4	<p>The module <b>TM2ALM3LT</b> in this setup needs to be configured as it is a multipurpose analog temperature module.</p> <p>Click on the upper half of the module to open the configuration.</p>																																	
5	<p>The I/O-mapping of the module is displayed in the lower part of this configuration screen.</p>	 <table border="1"><thead><tr><th>Used</th><th>Address</th><th>Symbol</th><th>Type</th><th>Scope</th><th>Minimum</th><th>Maximum</th><th>Units</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>%IW3.0</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr><tr><td><input type="checkbox"/></td><td>%IW3.1</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr><tr><td><input type="checkbox"/></td><td>%QW3.0</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr></tbody></table>	Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units	<input type="checkbox"/>	%IW3.0		Not in use	Normal	0	4095	None	<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None	<input type="checkbox"/>	%QW3.0		Not in use	Normal	0	4095	None
Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units																											
<input type="checkbox"/>	%IW3.0		Not in use	Normal	0	4095	None																											
<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None																											
<input type="checkbox"/>	%QW3.0		Not in use	Normal	0	4095	None																											
6	<p>In this example the <b>Type</b> of input is for both <b>PT100</b>, which you select from the drop-down list. The output remains unused.</p>	 <table border="1"><thead><tr><th>Used</th><th>Address</th><th>Symbol</th><th>Type</th><th>Scope</th><th>Minimum</th><th>Maximum</th><th>Units</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>%IW3.0</td><td></td><td>PT 100</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr><tr><td><input type="checkbox"/></td><td>%IW3.1</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr><tr><td><input type="checkbox"/></td><td>%QW3.0</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr></tbody></table>	Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units	<input type="checkbox"/>	%IW3.0		PT 100	Normal	0	4095	None	<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None	<input type="checkbox"/>	%QW3.0		Not in use	Normal	0	4095	None
Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units																											
<input type="checkbox"/>	%IW3.0		PT 100	Normal	0	4095	None																											
<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None																											
<input type="checkbox"/>	%QW3.0		Not in use	Normal	0	4095	None																											
7	<p>Select from the <b>Scope</b> drop-down list <b>Celsius</b>.</p>	 <table border="1"><thead><tr><th>Used</th><th>Address</th><th>Symbol</th><th>Type</th><th>Scope</th><th>Minimum</th><th>Maximum</th><th>Units</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>%IW3.0</td><td></td><td>PT 100</td><td>Celsius</td><td>-1000</td><td>5000</td><td>0.1 °C</td></tr><tr><td><input type="checkbox"/></td><td>%IW3.1</td><td></td><td>Not in use</td><td>Normal</td><td>0</td><td>4095</td><td>None</td></tr><tr><td><input type="checkbox"/></td><td>%QW3.0</td><td></td><td>Not in use</td><td>Customized</td><td>0</td><td>4095</td><td>None</td></tr></tbody></table>	Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units	<input type="checkbox"/>	%IW3.0		PT 100	Celsius	-1000	5000	0.1 °C	<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None	<input type="checkbox"/>	%QW3.0		Not in use	Customized	0	4095	None
Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units																											
<input type="checkbox"/>	%IW3.0		PT 100	Celsius	-1000	5000	0.1 °C																											
<input type="checkbox"/>	%IW3.1		Not in use	Normal	0	4095	None																											
<input type="checkbox"/>	%QW3.0		Not in use	Customized	0	4095	None																											

8	The analog channels are now configured in terms of input type including also the correct range.		Used	Address	Symbol	Type	Scope	Minimum	Maximum	Units
			<input type="checkbox"/>	%IW3.0		PT 100	Celsius	-1000	5000	0.1 °C
			<input type="checkbox"/>	%W3.1		PT 100	Celsius	-1000	5000	0.1 °C
			<input type="checkbox"/>	%QW3.0		Not in use	Normal	0	4095	None

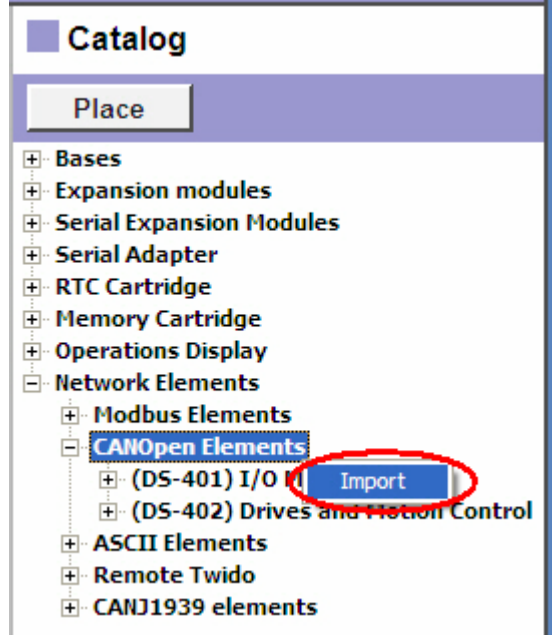
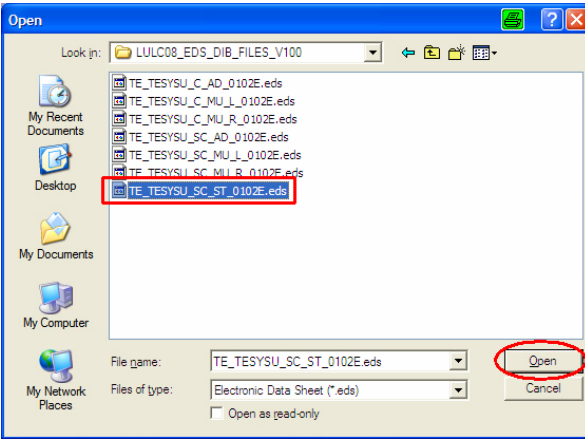
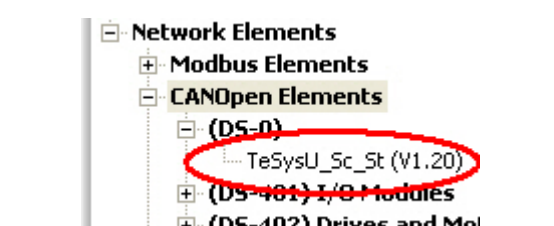
## Configure the Modbus for HMI

1	<p>Again in the <b>Catalog</b>, for the communications module, select:</p> <p><b>Network Elements-&gt;</b> <b>Modbus Elements-&gt;</b> <b>Magelis</b></p> <p>Pick and place it on the worktop.</p>	
2	<p>To configure the communications port for the HMI connection, right mouse click on the RS485-Port on the image of the Twido and select</p> <p><b>Configuration...</b></p> <p>in the pop-up menu.</p>	
3	<p>Select:</p> <p><b>Type:</b> Modbus</p> <p><b>Address:</b> 1</p> <p>Continue with <b>OK</b>.</p>	

<p>4 To configure the communication port of the HMI, right mouse click on the HMI image and select</p> <p><b>Configuration...</b></p> <p>in the pop-up menu.</p>	
<p>5 Insert:</p> <p><b>Name:</b> for example, the HMI type <b>HMISTU655</b></p> <p><b>Type: Modbus</b></p> <p><b>Address: Master</b></p> <p>Continue with <b>OK</b>.</p>	
<p>6 Now connect the HMI and the Twido with a line. Place the mouse at the end of the Twido port; the view changes. Hold the right mouse button and draw a line to the point below the HMI image.</p> <p>Release the mouse button and the connection, <b>my network 1</b> is shown. Double click on:</p> <p><b>my network 1</b></p> <p>to open the configuration box.</p>	

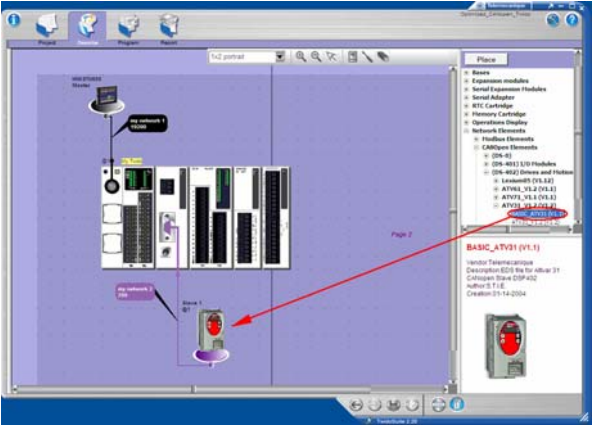
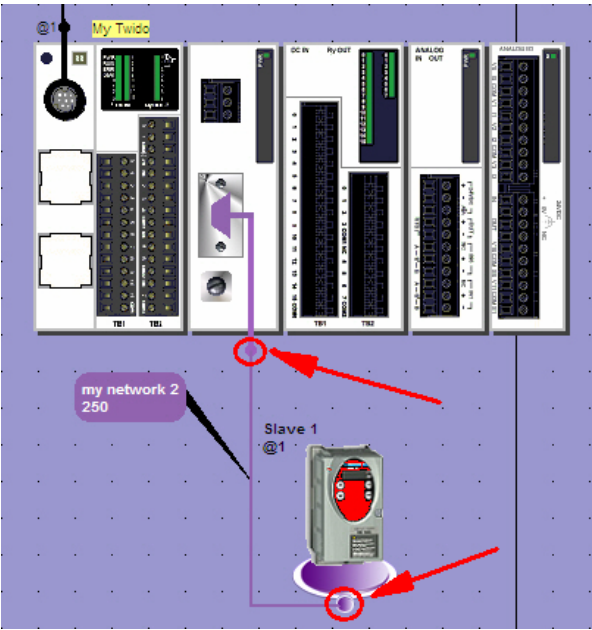
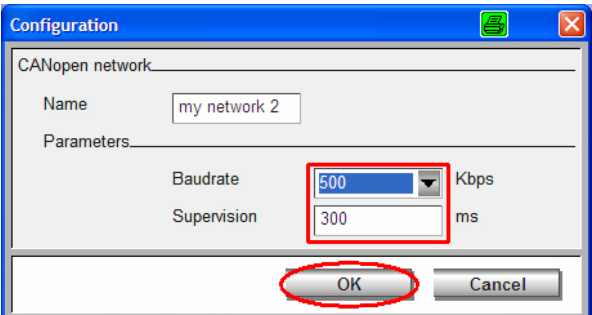
7	<p>Use the following parameter values:</p> <p><b>Baudrate: 19200</b>  <b>Data Bits: 8(RTU)</b>  <b>Parity: none</b>  <b>Stop Bit: 1</b></p> <p>Confirm with <b>OK</b>.</p>	
---	--	--

## Import EDS-file for TeSysU

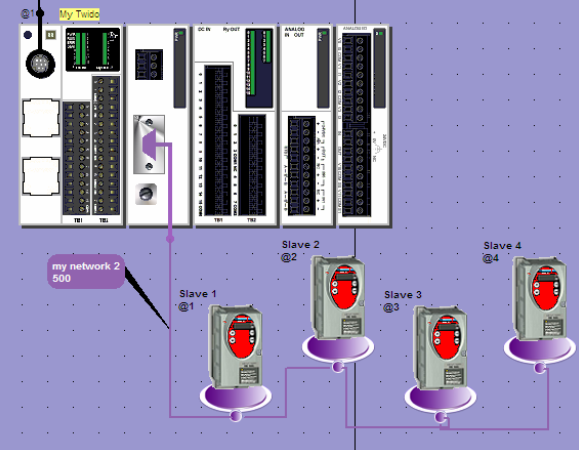
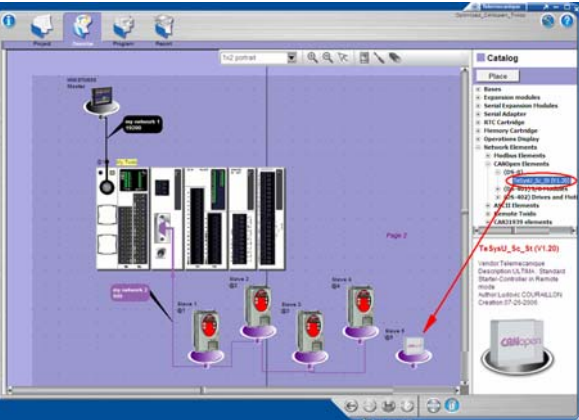
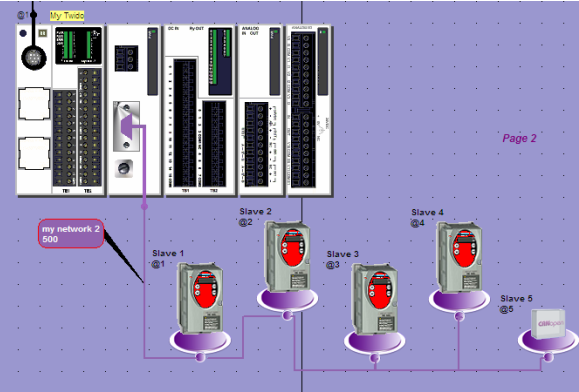
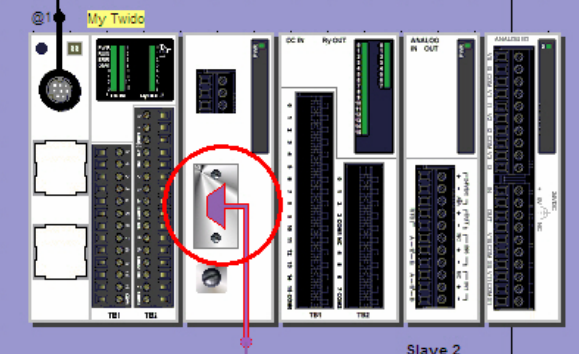
1	<p>If a device is not existent in the catalog, you must import the needed eds-file for the CANopen configuration.</p> <p>Right mouse click on</p> <p><b>CANopen elements -&gt; Import</b></p>	
2	<p>Select the corresponding file in the project folder</p> <p><b>TESYSU_SC_ST_0102E.eds</b></p>	
3	<p>The new device is now available in the catalog.</p>	

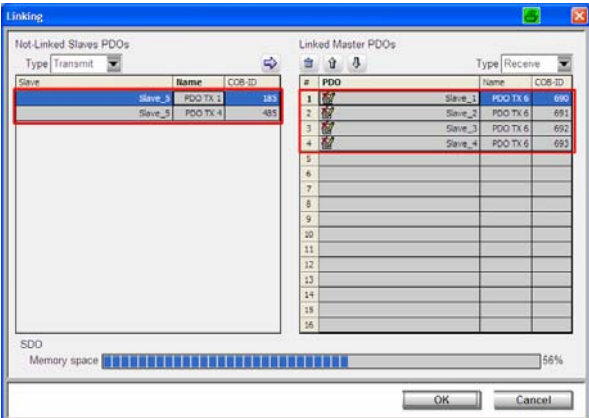
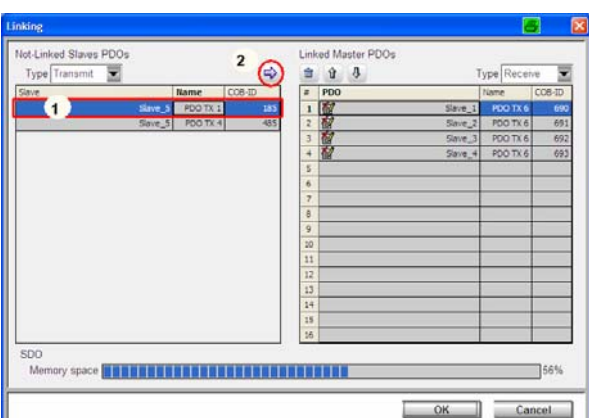
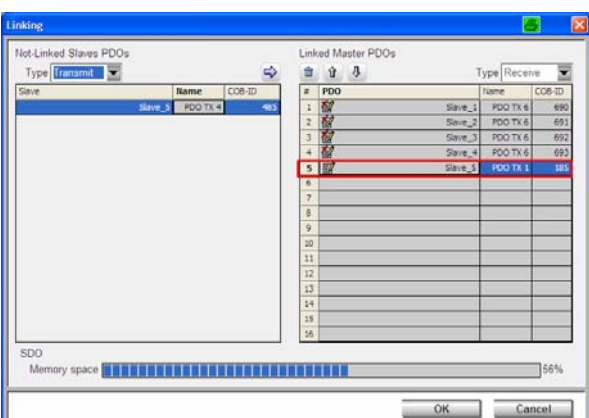
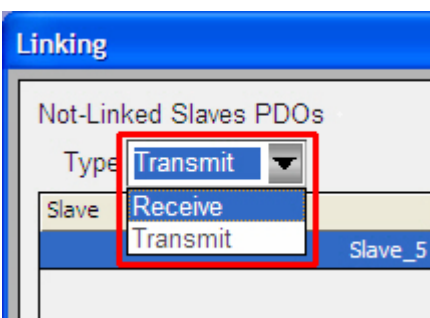


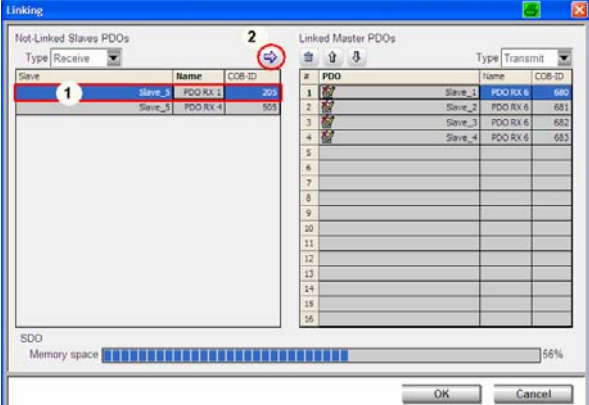
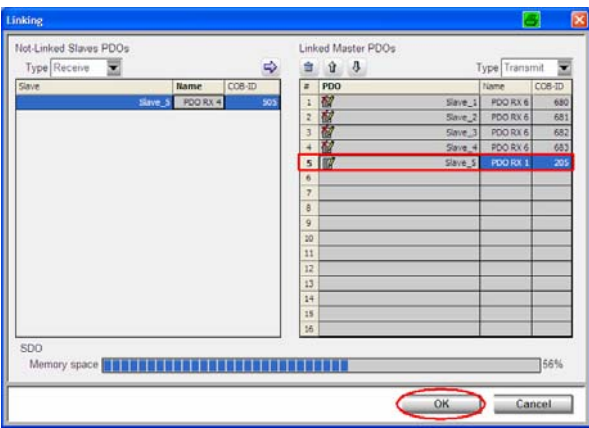
## Configure the CANopen devices

1	<p>First configure the hardware, then activate and define the slaves.</p> <p>Without activation this slave is only shown in the hardware configuration, highlighted in red, but the CANbus communication is inactive. This could be helpful for example, to test a single drive of a machine.</p>
2	<p>The next step is to add the CANopen slaves.</p> <p><b>Network Elements</b>  → CANopen Elements  → (DS-402) Drives and Motion Control  → ATV31_V1.2  → Basic_ATV31</p> <p>Pick and drop at workspace:  <b>4x ATV31</b>  <b>1x TeSysU</b> (see next chapter)</p> <p><b>Note;</b>  Altivar 312 is fully compatible with the Altivar 31 regarding mechanical characteristics and software. The ATV31 devices shown and described in this example are already integrated in Twido delivery and software installation by default.</p> 
3	<p>Connect the CANopen master module and the slave with a line.</p> <p>Place the mouse at the end of the CANopen port; it changes the view. Hold the left mouse button and draw a line to the point below the slave image.</p> <p>Release the mouse button and the connection <b>my network 2</b> is shown.</p> <p>Double click on:</p> <p><b>my network 2</b></p> <p>to open the configuration dialog box.</p> 
4	<p>In this application following parameters are used:</p> <p><b>Baudrate: 500 Kbps</b>  <b>Supervision: 300 ms</b></p> <p>Confirm with <b>OK</b>.</p> 

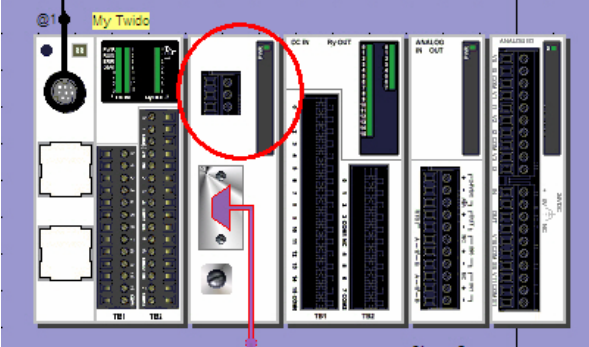


5	<p>Repeat <b>steps 2 &amp; 3</b> to add the next ATV312 slaves.</p> <p>Connect <b>slave 1, 2, 3, 4</b> with a line.</p>	
6	<p>To add the TeSysU click in the catalog on:</p> <p><b>Network Elements</b>  → <b>CANOpen Elements</b>  → <b>(DS-0)</b>  → <b>TeSysU_Sc_St (V1.20)</b></p>	
7	<p>Now connect the TeSysU to the last Slave.</p>	
8	<p>Double click on the CANopen connector image of the CANopen module to open up the linking configuration.</p>	

9	<p>The <b>Linking</b> dialog box shows you on the right hand side the PDOs (Process Data Objects) which have already been linked (Altivar slaves 1..4) and on the left hand side the ones which have not been linked so far (TeSysU slave 5).</p>	
10	<p>To convert a not-linked to a linked PDO:</p> <ol style="list-style-type: none"> <li>(1) Click the first <b>PDO TX 1</b></li> <li>(2) Click on the arrow symbol at the top</li> </ol>	
11	<p>The <b>PDO TX 1</b> appears now in the list of linked PDOs on the right.</p> <p><b>Note:</b> This PDO transmits the TeSysU module status to the Twido.</p>	
12	<p>Change now to the receive PDOs by selecting <b>Receive</b> from the <b>Type</b> drop-down list.</p>	

13	<p>Proceed in the same way as you did with the transmit PDOs:</p> <ol style="list-style-type: none"> <li>(1) Click the first <b>PDO RX 1</b></li> <li>(2) Click on the arrow symbol at the top</li> </ol>	
14	<p>The <b>PDO RX 1</b> appears now in the list of linked PDOs on the right.</p> <p><b>Note:</b> This PDO receives the control word from the Twido.</p> <p>Close the <b>Linking</b> dialog box by clicking <b>OK</b>.</p>	

## CANopen module configuration

1	<p>To open directly the CANopen module configuration click on the upper half of the CANopen module.</p>	
---	---	--

2 Alternatively you can go to the **Module configuration** with:

- (1) Program
- (2) Configure
- (3) Configure the Hardware

**Slave 1** is highlighted in the figure. The table on the right shows the addresses and the descriptions of the default parameters.

Drive macros for the Altivar devices have already been created automatically and are displayed in the table at the bottom.

#	Slave	Type	Supervision	Set	Used	Address	Symbol	Object	Size
1	Slave_1	BASIC ATVIS (V1.1)	None	Default	<input checked="" type="checkbox"/>	%IWC1.0.0	D_STATUS_SLAVE_1	Drivecom status register;	16
2	Slave_2	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%IWC1.0.1	D_CONTROL_SLAVE_1	Control effort;	16
3	Slave_3	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%IWC1.0.2	D_IERROR_SLAVE_1	Error code;	16
4	Slave_4	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%IWC1.0.3	D_MCURRENT_SLAVE_1	Motor Current;	16
5	Slave_5	TaSynULS_2B (V1.2B)	None	Default	<input type="checkbox"/>	%QWC1.0.0	D_COMMAND_SLAVE_1	Drivecom command reg.;	16
					<input type="checkbox"/>	%QWC1.0.1	D_TARGET_SLAVE_1	Target velocity;	16

3 The **Used** column indicates whether it is used in the programming code. The **Address** is the variable address in programming code, **Symbol** is the symbolic name and **Object** is a description.

Used	Address	Symbol	Object	Size
<input type="checkbox"/>	%IWC1.0.0	D_STATUS_SLAVE_1	Drivecom status register;	16
<input type="checkbox"/>	%IWC1.0.1	D_CONTROL_SLAVE_1	Control effort;	16
<input type="checkbox"/>	%IWC1.0.2	D_IERROR_SLAVE_1	Error code;	16
<input type="checkbox"/>	%IWC1.0.3	D_MCURRENT_SLAVE_1	Motor Current;	16
<input type="checkbox"/>	%QWC1.0.0	D_COMMAND_SLAVE_1	Drivecom command reg.;	16
<input type="checkbox"/>	%QWC1.0.1	D_TARGET_SLAVE_1	Target velocity;	16

4 By clicking on Slave\_5 you find the corresponding objects and addresses in the list on the right.

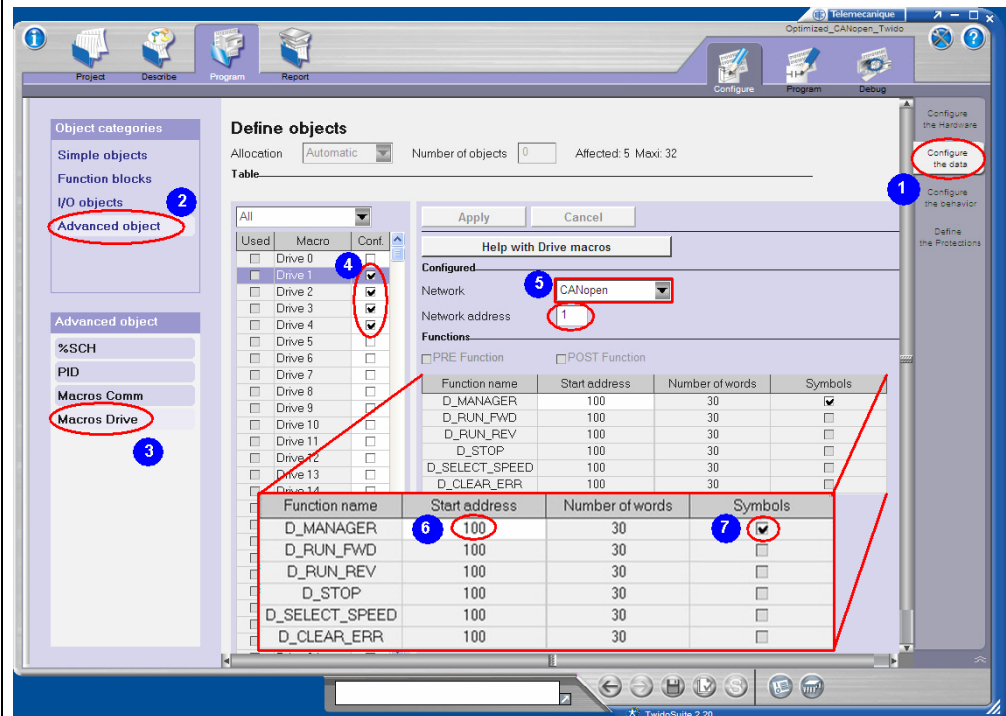
#	Slave	Type	Supervision	Set	Used	Address	Symbol	Object	Size
1	Slave_1	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%IWC1.4.0	TPDO_1	Status register	16
2	Slave_2	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%IWC1.4.1		I/O module status register	16
3	Slave_3	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%QWC1.4.0		Control of the system	16
4	Slave_4	BASIC ATVIS (V1.1)	None	Default	<input type="checkbox"/>	%QWC1.4.1		Control of comm module	16
5	Slave_5	TaSynULS_2B (V1.2B)	None	Default	<input type="checkbox"/>	%QWC1.4.2		Control of outputs	16

5 In this list you can type in your own Symbols for a clear identification in your program.

Used	Address	Symbol	Object	Size
<input type="checkbox"/>	%IWC1.4.0	TPDO_1	Status register	16
<input type="checkbox"/>	%IWC1.4.1		I/O module status register	16
<input type="checkbox"/>	%QWC1.4.0		Control of the system	16
<input type="checkbox"/>	%QWC1.4.1		Control of comm module	16
<input type="checkbox"/>	%QWC1.4.2		Control of outputs	16

## Activate the Altivar drive macros

- 1 Click on:
  - (1) Configure the data
  - (2) Advanced objects
  - (3) Macros Drive
  - (4) Activate the slave in the column Conf. and
  - (5) select CANopen as Network with the respective Network Address (1..4).



- (6) In the column **Start address** enter for this example 100, 200, 300, 400 for the ATV drives. Confirm with Enter. The predefined length is a block of 30 words for each drive.
- (7) Mark the checkbox **Symbols** to get the predefined tags for the words used by the macro.

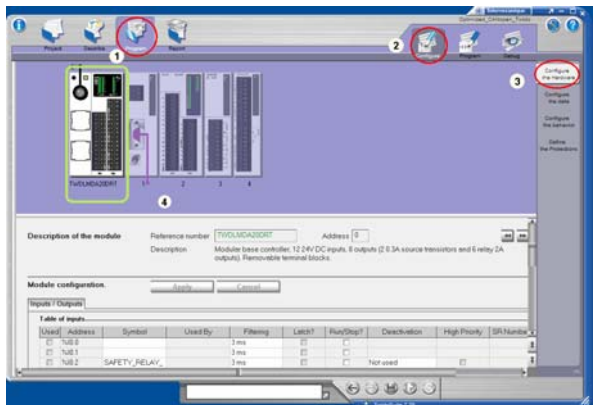
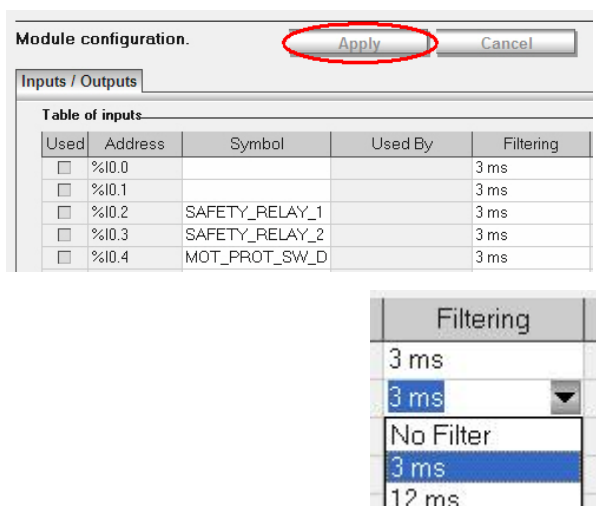
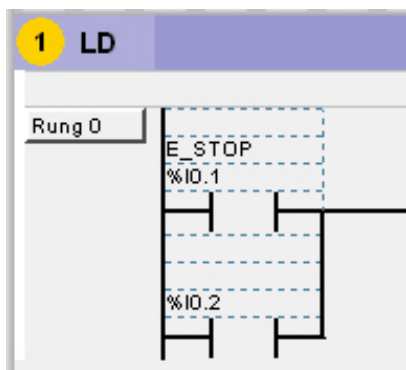
- 2 The slaves are now configured and ready for use in the programming code.

## Create Variables

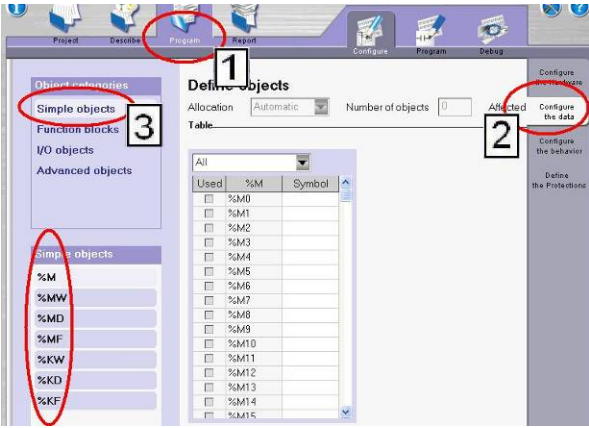
Two different variable types are possible:

- I/O variables, linked to inputs or outputs
- internal variables, such as bit, byte, word, double word, floating point and constant

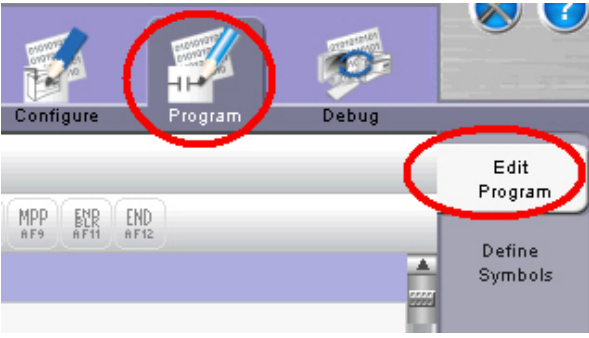
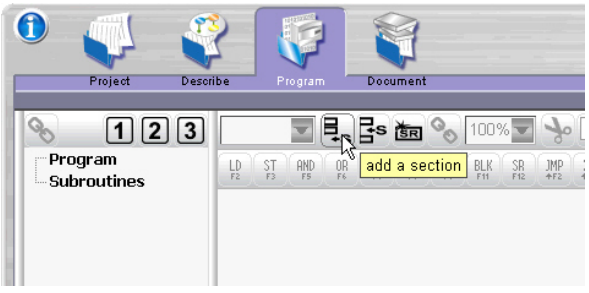
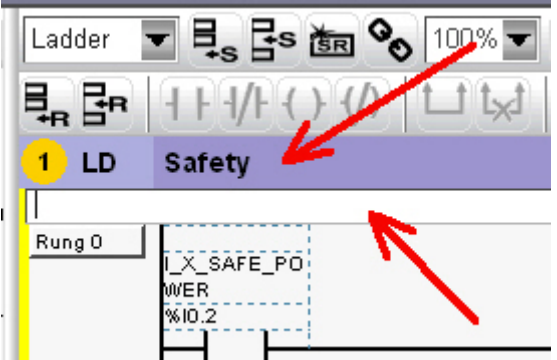
The variables can be defined in separate lists or entered directly in the programming editor (list or ladder).

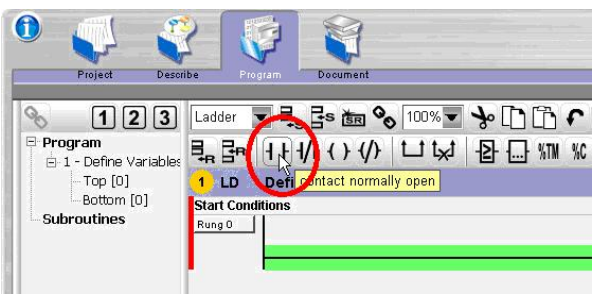
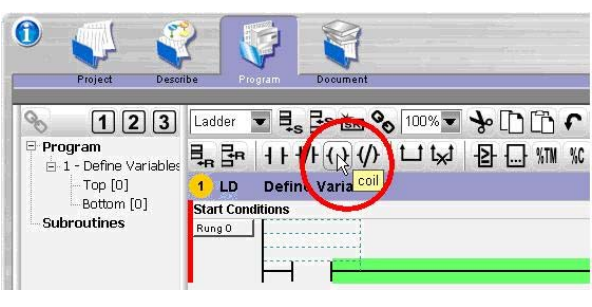
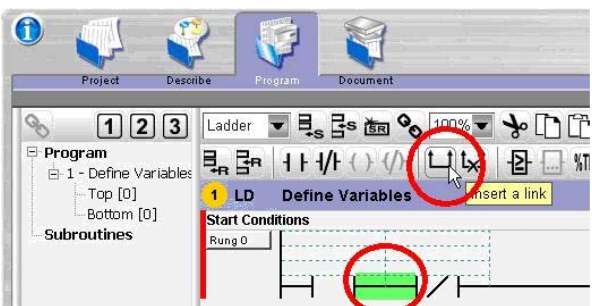
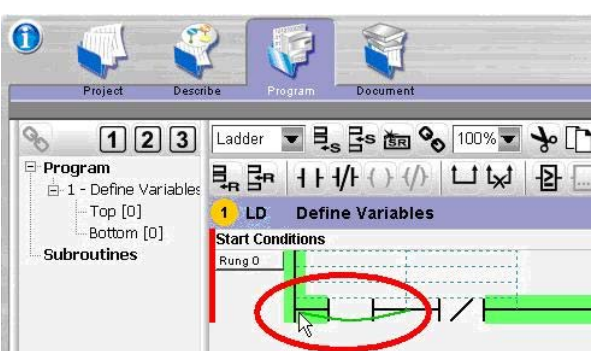
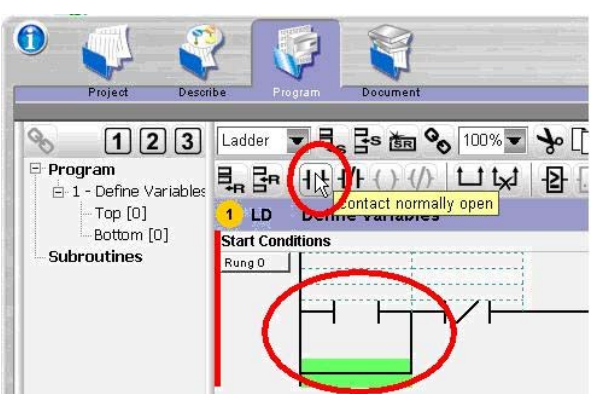
<p><b>1</b></p> <p>To create I/O variables, open the <b>Module configuration</b> with</p> <ol style="list-style-type: none"> <li>(1) <b>Program</b></li> <li>(2) <b>Configure</b></li> <li>(3) <b>Configure the Hardware</b></li> <li>(4) Then click the relevant I/O module.</li> </ol> <p>The table at the bottom of the screen shows the I/O variables.</p>																															
<p><b>2</b></p> <p>The following is shown in the table:</p> <p><b>Used:</b> indicates whether the variable is used in the programming code</p> <p><b>Address:</b> the I/O address</p> <p><b>Symbol:</b> the symbolic variable</p> <p><b>Used By:</b> user logic or empty</p> <p><b>Filtering:</b> minimum signal time length (only for inputs)</p> <p>Confirm with <b>Apply</b>.</p>	 <table border="1" data-bbox="869 1008 1444 1153"> <caption>Table of inputs</caption> <thead> <tr> <th>Used</th> <th>Address</th> <th>Symbol</th> <th>Used By</th> <th>Filtering</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>%I0.0</td> <td></td> <td></td> <td>3 ms</td> </tr> <tr> <td><input type="checkbox"/></td> <td>%I0.1</td> <td></td> <td></td> <td>3 ms</td> </tr> <tr> <td><input type="checkbox"/></td> <td>%I0.2</td> <td>SAFETY_RELAY_1</td> <td></td> <td>3 ms</td> </tr> <tr> <td><input type="checkbox"/></td> <td>%I0.3</td> <td>SAFETY_RELAY_2</td> <td></td> <td>3 ms</td> </tr> <tr> <td><input type="checkbox"/></td> <td>%I0.4</td> <td>MOT_PROT_SW_D</td> <td></td> <td>3 ms</td> </tr> </tbody> </table>	Used	Address	Symbol	Used By	Filtering	<input type="checkbox"/>	%I0.0			3 ms	<input type="checkbox"/>	%I0.1			3 ms	<input type="checkbox"/>	%I0.2	SAFETY_RELAY_1		3 ms	<input type="checkbox"/>	%I0.3	SAFETY_RELAY_2		3 ms	<input type="checkbox"/>	%I0.4	MOT_PROT_SW_D		3 ms
Used	Address	Symbol	Used By	Filtering																											
<input type="checkbox"/>	%I0.0			3 ms																											
<input type="checkbox"/>	%I0.1			3 ms																											
<input type="checkbox"/>	%I0.2	SAFETY_RELAY_1		3 ms																											
<input type="checkbox"/>	%I0.3	SAFETY_RELAY_2		3 ms																											
<input type="checkbox"/>	%I0.4	MOT_PROT_SW_D		3 ms																											
<p><b>3</b></p> <p>The image shows how the variables in the table above are used in the program:</p> <p><b>%I0.1 – E_STOP:</b> variable and symbolic name in program code</p> <p><b>%I0.2:</b> variable without symbolic name in program code</p> <p><b>%I0.3 - START:</b> the symbolic name is in the variable list (see image above) but not used in the program.</p>																															



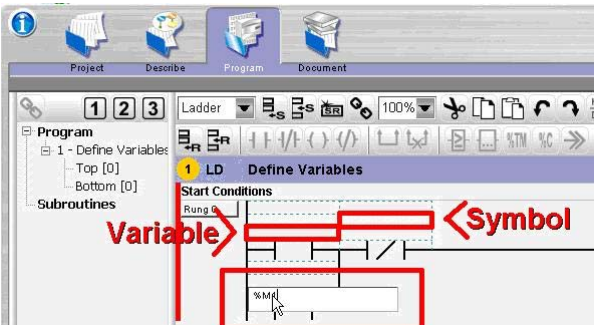
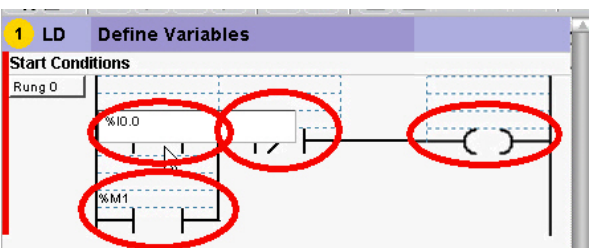
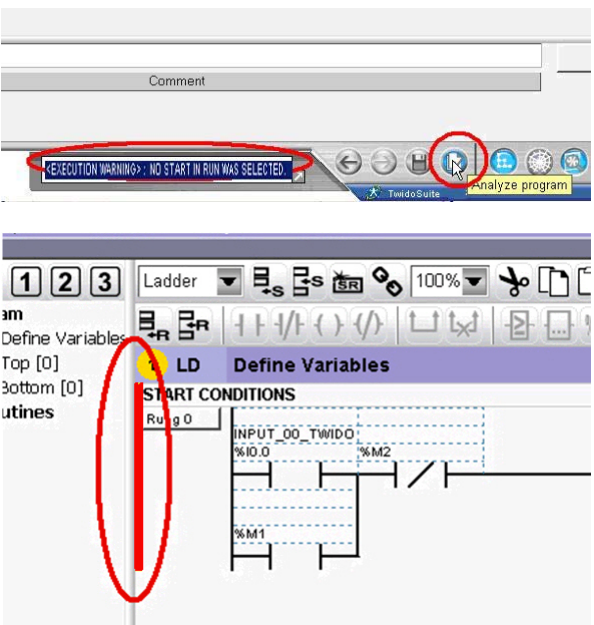
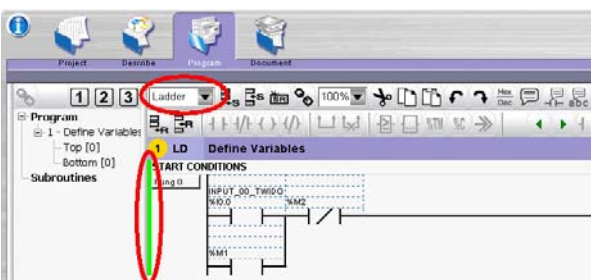
<p><b>4</b></p>	<p>To create internal variables, open the Module configuration:</p> <p><b>Program-&gt;Configure the Data-&gt;Simple Objects</b></p> <p>The table, bottom left, lists the datatypes: <b>%M</b>, <b>%MW</b>, <b>%MD</b>, etc.</p> <p>A table with addresses and symbolic names for each data type is shown on the right.</p> <p><b>Used:</b> indicates whether the variable is used in the program</p> <p><b>%Mx:</b> variable address</p> <p><b>Symbol:</b> symbolic variable</p>	
-----------------	--	--


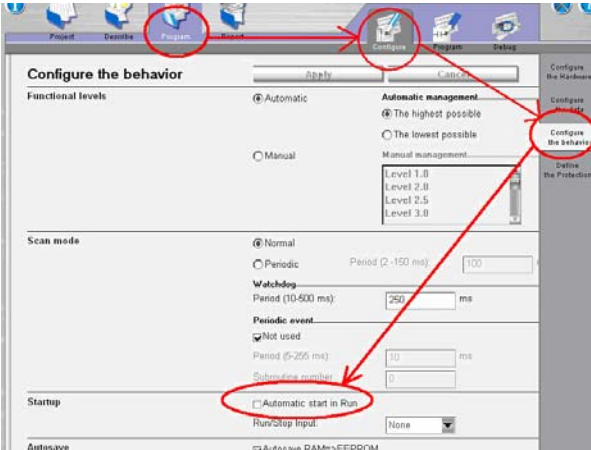
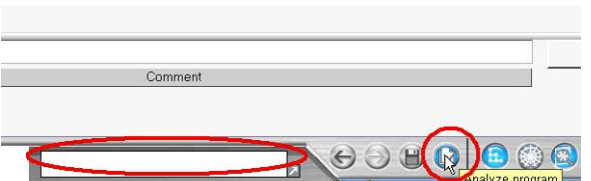
## Create an Application Program

<p><b>1</b></p>	<p>Go to the program editor with:</p> <p><b>Program-&gt;Edit Program</b></p> <p>A new program always begins with <b>LADDER</b> as the programming language.</p> <p>You can only view the program in LIST mode once the first rung has been completed and analyzed.</p>	
<p><b>2</b></p>	<p>Start the programming with:</p> <p><b>add a section.</b></p> <p>This creates a section with empty rungs.</p>	
<p><b>3</b></p>	<p>Click on the header of a <b>Section</b> to enter a description (for example <b>"Safety"</b>).</p> <p>Click on the header of a rung to enter a description.</p>	

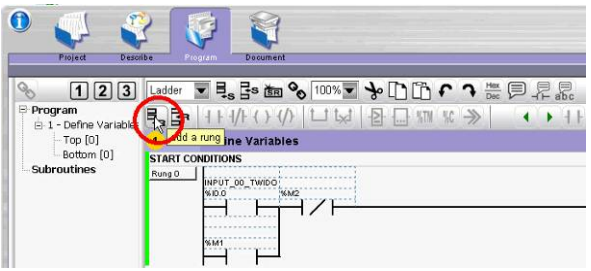
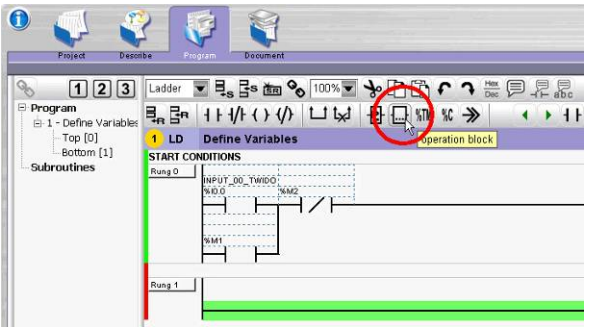
<p><b>5</b></p>	<p>To add contacts, click on the rung. The rung changes its color and indicates the current rung is being edited.</p> <p>Clicking on an icon in the toolbar inserts the selected object at the first available place on the rung.</p>	
<p><b>6</b></p>	<p>Add coils in the same manner.</p>	
<p><b>7</b></p>	<p>To make a logical connection, simply mark the source and pull it to the destination. All possible connecting points are shown in same color.</p>	
<p><b>8</b></p>	<p>Once a link has been inserted, you can add a contact to form a logical OR.</p>	
<p><b>9</b></p>	<p>To assign an object to an address or variable, click on the upper half of the object.</p>	

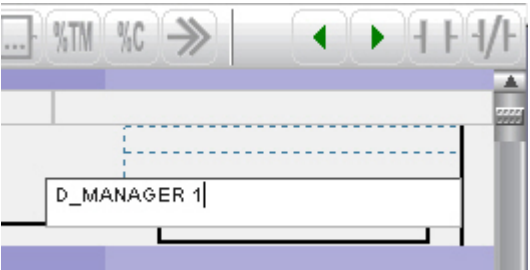
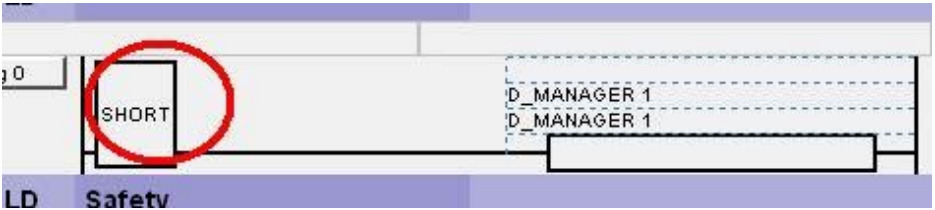


10	<p>The editor automatically recognizes whether a rung is syntactically complete and marks it in yellow.</p> <p>The rung, however, has not yet been analyzed.</p>	
11	<p>The analysis is done for all contacts.</p>	
12	<p>Analyze the program code by clicking on the icon in the bottom right hand corner of the window.</p> <p>The message box to the left of the icon shows if there are any compilation errors.</p>	
13	<p>If the analysis is successful, the red band changes to green.</p>	

14	<p>If compilation errors are found, they are listed at the bottom of the edit field. Click on the arrow icon on the right side of the message box to enlarge it.</p> <p>A common compilation message may be:</p> <p><b>No Start In Run Was Selected</b></p>	
15	<p>To fix this, select:</p> <p><b>Program-&gt; Configure-&gt; Configure the behavior -&gt;</b></p> <p>and for <b>Startup</b> activate:</p> <p><b>Automatic start in Run</b></p> <p>This is an operation mode that causes the Twido to start up automatically on return of power after a power interruption.</p>	
16	<p>After restarting the analysis, the notification field is empty.</p>	

## Insert Macros for ATV Drives

1	<p>To use the macros for the drives, a second Rung is added using the <b>add a rung</b> function.</p>	
2	<p>Insert an operation block for the macro.</p>	

<p><b>3</b></p>	<p>To link the operation block to the macro drive input <b>D_manager 1</b>. D_manager is the macro function to access the drive and the 1 indicates Drive 1.</p> <p>The space between D_manager and the 1 is mandatory. Since the macro is already configured, the address field also contains the symbolic macro name <b>D_MANAGER 1</b>. The <b>SHORT</b> (see next step) must be exchanged for a control variable.</p> <p>You can now repeat this for the other macros:</p> <p><b>D_CLEAR_ERR</b> <b>D_RUN_FWD</b> <b>D_RUN_REV</b> <b>D_STOP</b> <b>D_SELECT_SPEED</b></p>	
<p><b>4</b></p>		

## Control commands for TeSysU

- 1** For TeSysU no specific macro are required. The commands forward, reverse and stop can be controlled by using the linked PDOs,
- The PDO mapping for all TeSysU modules are the same, but not all modules use or provide the bits which are predefined in these transfer words.
- The TeSys U Sc St module in this example (standard starter controller module) provides as status to the Twido
- status register
  - I/O module status register
- The TeSysU module commands for start and stop are controlled only by the first transfer word (**Word 1**). Forward and reverse command are bitwise controlled. If both bits are switched back to zero (low signal) TeSysU module stops.
- Note:**  
The graphics on the right show only the used bits within the CANopen objects for this type of module.

- 2** You find the correct input and output word for control and status of the TeSysU by opening the CANopen module configuration.
- According to this example the address of the first output word is %QWC1.4.0.

### Statuswords: TeSysU → Twido

		Word 1	Word 2
TeSys U Sc St	Register	455	458
	CANopen Index	2004:6	2004:9
	Description	Status register	I/O module status register

Word 1	CANopen Index 2004:6	Sc St
bit 0	Ready: LUB**/2B** = the rotary handle is turned to 'On' position and there is no fault. LUS**/2S** = the push-button is pressed and there is no fault.	✓
bit 1	Pole status: closed	✓
bit 2	All faults	✓
bit 3	All warnings	✓
bit 4	Tripped: LUB**/2B** = the rotary handle is turned to 'Trip' position. LUS**/2S** = the push-button is depressed.	✓
bit 14	(Non significant)	✓

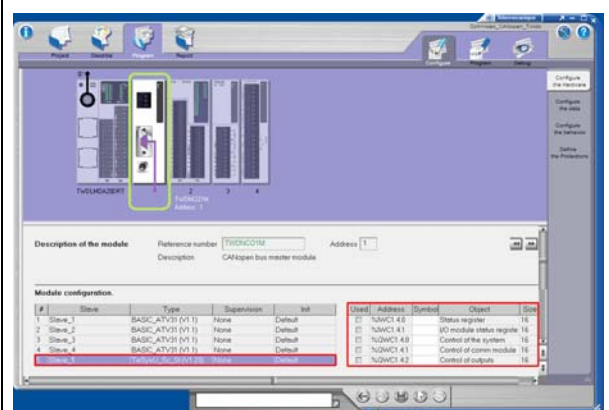
Word 2	CANopen Index 2004:9	Sc St
bit 0	OA1 status	✓
bit 1	OA3 status	✓
bit 2	LO1 status	✓
bit 8	LI1 status	✓
bit 9	LI2 status	✓

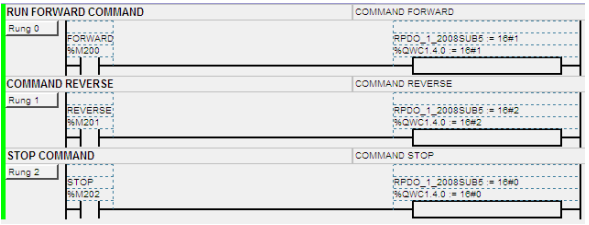
### Control words: Twido → TeSysU

	Word 1	Word 2	Word 3	Word 4
Register	704	703	700	Empty
CANopen Index	2008:5	2008:4	2008:1	—
Description	Control Register	Control of communication module	Output control	—

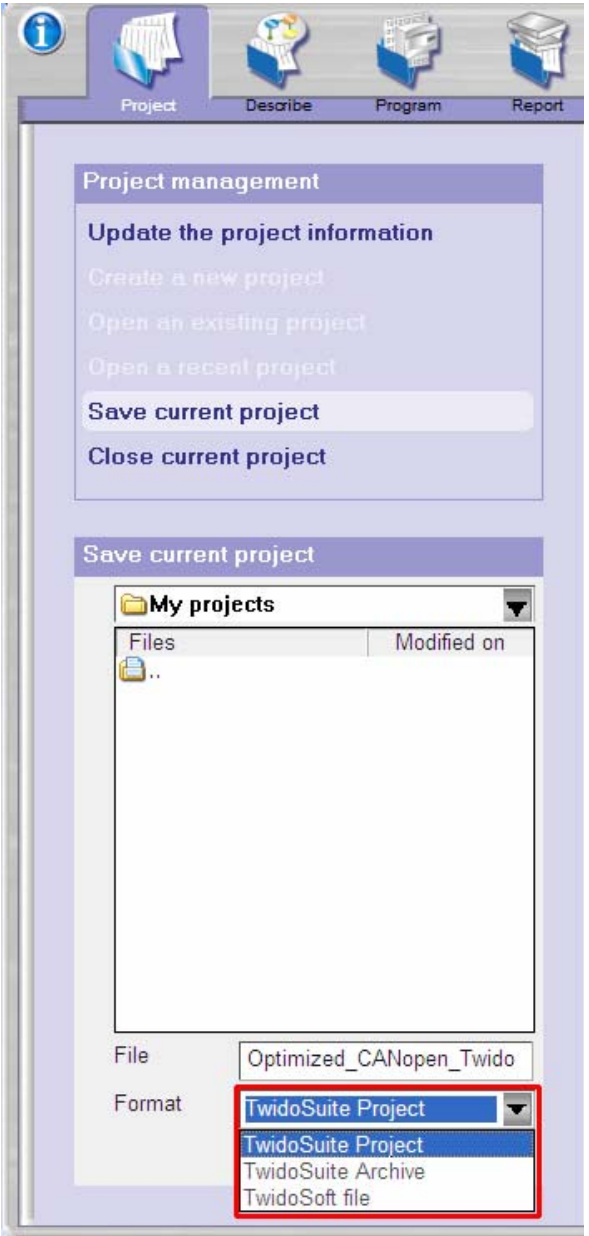
Word 1	CANopen Index 2008:5	Sc St
bit 0	Run forward	✓
bit 1	Run reverse	✓

Word 2	CANopen Index 2008:4	Sc St
bit 3	Reset warning (communication loss,...) This bit is active on rising edge and must be reset to 0 by programming	✓



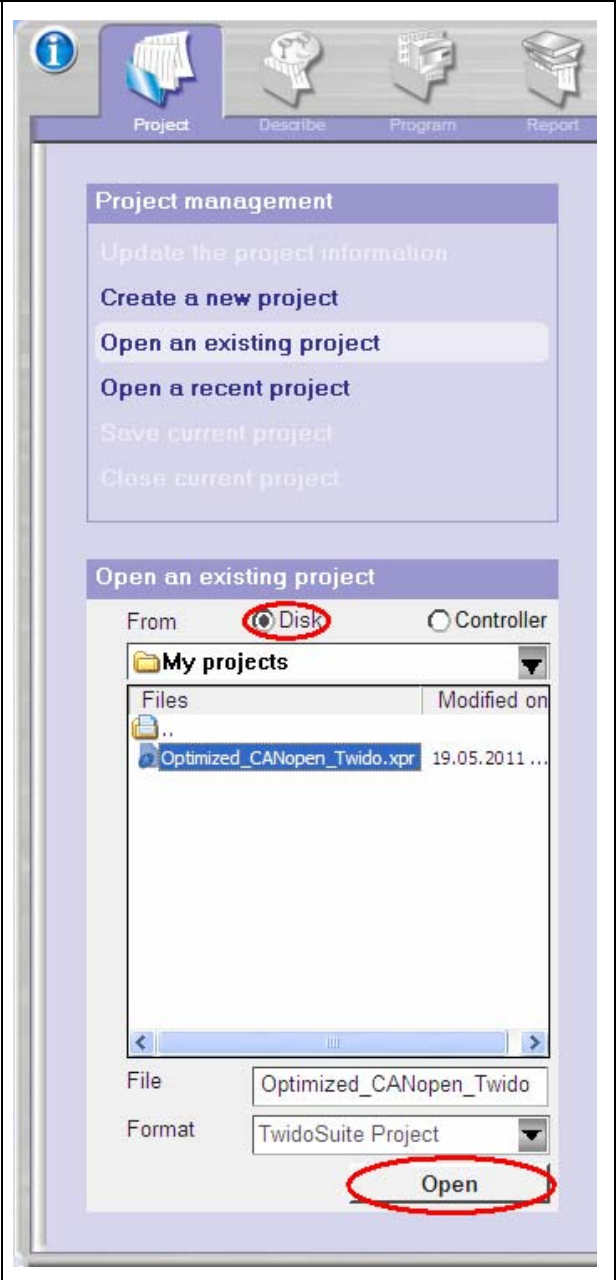
3	<p>To write the commands to the TeSysU the control word has to be provided with only one command active at a time with maximum one bit set. Put following equation in the rung of your program to achieve the respective command:</p> <p style="text-align: right;"> <b>Forward → 16#1</b>  <b>Reverse → 16#2</b>  <b>Stop → 16#0</b> </p>	
---	--	--

## Save / Export a project

1	<p>To save the project select <b>Save current project</b> in the project menu.</p> <p>In the <b>Save current project</b> window you can select the folder for the save, input the <b>File</b> name and select <b>TwidoSuite Project</b> from the <b>Format</b> drop-down list.</p> <p><b>Note:</b> You can save the file also as an archive file. The archive file has the advantage of being portable and is a single file.</p>	
---	--	---

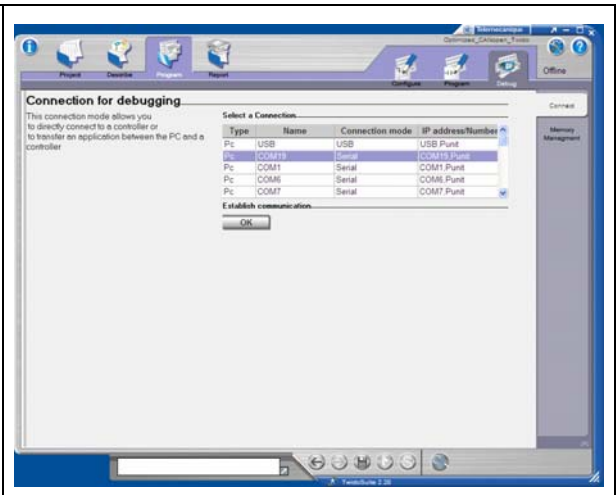
## Open / import a project

- 1 To open a project use the **Open an existing project** function in the project menu.  
  
In the **open an existing project** window select the medium:  
  
**Disk or Controller**  
  
by clicking the radio button.  
  
Choose the file type under **Format** (for a project file: **Project**), select the folder and select the **File** you wish to open.  
  
When ready click on **Open** to open the project.  
  
For import the action is the same, except the file **Format** is TwidoSuite Archive.



## Connecting to the Twido

- 1 To connect the PC to the Twido select **Program** in the main menu, then **Debug** in the sub-menu to get access to the **Connect** selection.





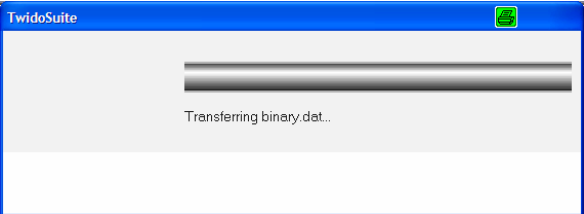
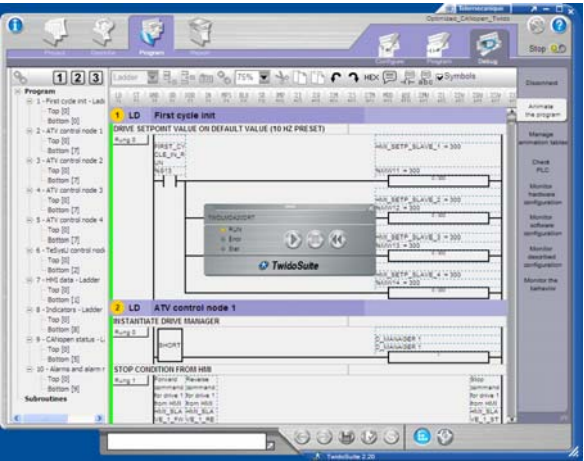

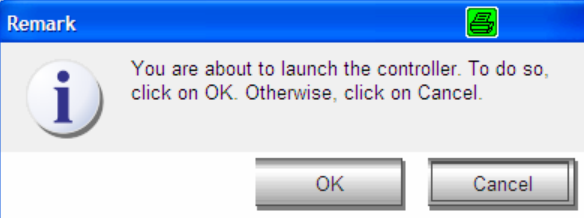

<div>2</div>	<div>The available communication ports are shown.</div> <div>Check that the the TSXCUSB485 adapter is connected, and select the <b>linked COM port</b>.</div> <div>Continue with <b>OK</b>.</div> <div><b>Note:</b> If you don't know which COM port is assigned to the TSXCUSB485 adapter continue with step 3. If you already know the COM port go directly to step 5.</div>	<div><div>Select a Connection</div><table><thead><tr><th>Type</th><th>Name</th><th>Connection mode</th><th>IP address/Number</th></tr></thead><tbody><tr><td>Pc</td><td>USB</td><td>USB</td><td>USB,Punit</td></tr><tr><td>Pc</td><td>COM19</td><td>Serial</td><td>COM19,Punit</td></tr><tr><td>Pc</td><td>COM1</td><td>Serial</td><td>COM1,Punit</td></tr><tr><td>Pc</td><td>COM6</td><td>Serial</td><td>COM6,Punit</td></tr><tr><td>Pc</td><td>COM7</td><td>Serial</td><td>COM7,Punit</td></tr></tbody></table><div>Establish communication</div><div>OK</div></div>	Type	Name	Connection mode	IP address/Number	Pc	USB	USB	USB,Punit	Pc	COM19	Serial	COM19,Punit	Pc	COM1	Serial	COM1,Punit	Pc	COM6	Serial	COM6,Punit	Pc	COM7	Serial	COM7,Punit
Type	Name	Connection mode	IP address/Number																							
Pc	USB	USB	USB,Punit																							
Pc	COM19	Serial	COM19,Punit																							
Pc	COM1	Serial	COM1,Punit																							
Pc	COM6	Serial	COM6,Punit																							
Pc	COM7	Serial	COM7,Punit																							
<div>3</div>	<div>The TSXCUSB485 adapter is linked by Windows to a virtual COM port. To identify the right COM port you need to open the <b>System Properties</b> dialog by clicking on <b>Windows key + Pause</b> button.</div> <div>Open up the <b>Hardware</b> tab and click on <b>Device Manager</b>.</div>	<div><div>System Properties</div><div>AdvancedSystem RestoreAutomatic Updates</div><div>GeneralComputer NameHardware</div><div><div>Device Manager</div><div>The Device Manager lists all the hardware devices installed on your computer. Use the Device Manager to change the properties of any device.</div><div>Device Manager</div></div><div><div>Drivers</div><div>Driver SigningWindows Update</div></div><div><div>Hardware Profiles</div><div>Hardware Profiles</div></div><div>OKCancelApply</div></div>																								
<div>4</div>	<div>In the <b>Device Manager</b> click on <b>Ports (COM &amp; LPT)</b>.</div> <div>The <b>TSX C USB 485</b> adapter is listed with the corresponding port in brackets. In this case the port <b>COM19</b> is assigned to it.</div>	<div><div>Device Manager</div><div>FileActionViewHelp</div><div>KeyboardsMice and other pointing devicesModemsMonitorsNetwork adaptersPCMCIA adaptersPortable DevicesPorts (COM &amp; LPT)BT Port (COM10)BT Port (COM11)BT Port (COM12)BT Port (COM13)BT Port (COM14)BT Port (COM20)BT Port (COM21)BT Port (COM22)BT Port (COM6)BT Port (COM7)Communications Port (COM1)ECP Printer Port (LPT1)TSX C USB 485 (COM19)</div></div>																								

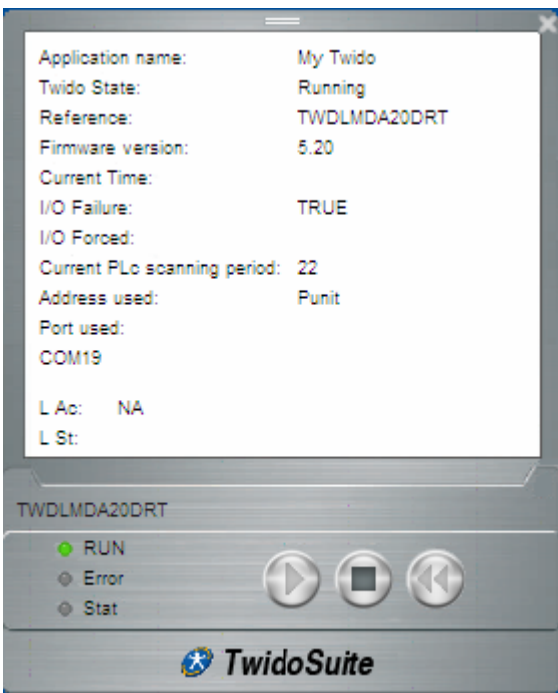
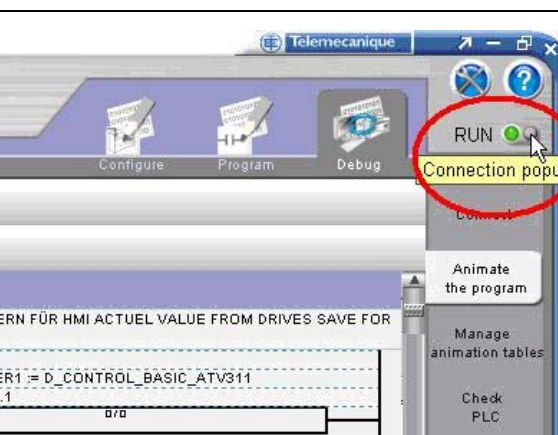
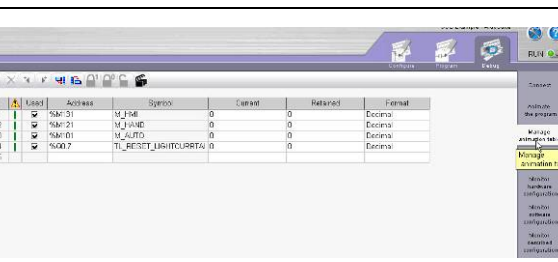
5	A progress bar is displayed while TwidoSuite establishes the connection.	
---	--	--

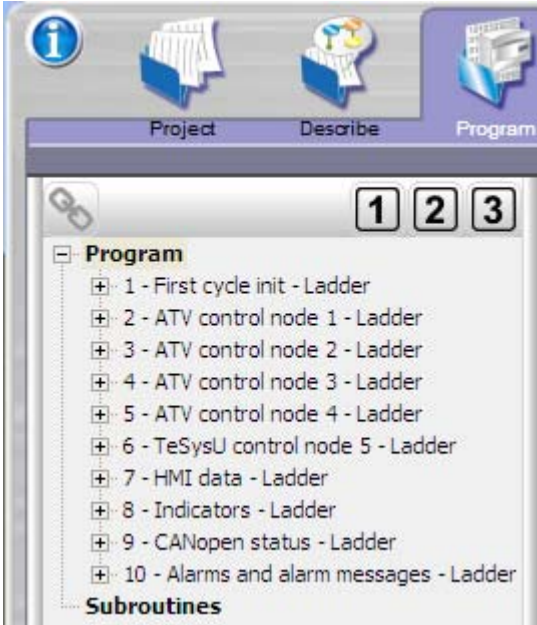
## Download a project

1	<p>After successfully connecting to the Twido, TwidoSuite compares the project in the Twido and the opened project.</p> <p>You can now either download your project or upload the contents of the Twido</p>	
2	<p>To download the project select:</p> <p><b>Transfer PC ==&gt; controller</b></p> <p>and confirm with <b>OK</b>.</p>	
3	<p>If the Twido is already running it must be stopped first.</p> <p>Confirm with <b>OK</b> if you wish to proceed.</p>	
4	<p>You will be informed that the contents of the Twido are about to be overwritten.</p> <p>Confirm with <b>OK</b> to continue.</p>	





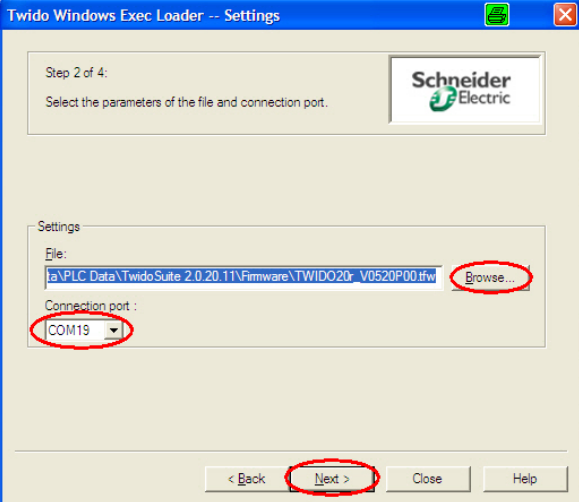
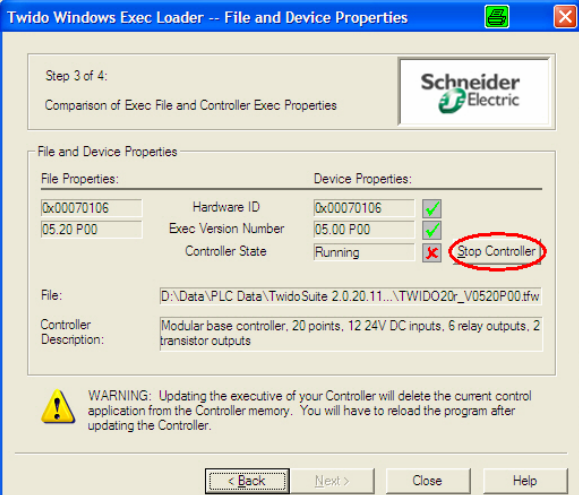
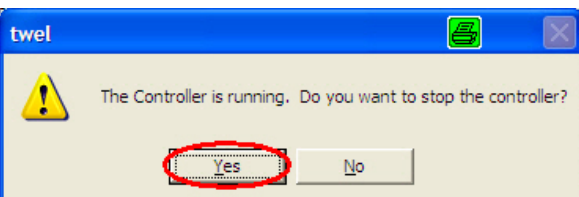
5	<p>The download status is indicated with a process bar.</p>	
6	<p>After a successful download, TwidoSuite changes to online viewing mode.</p>	
7	<p>A control panel is shown so you can start and stop the Twido.</p> <p>The upper button extends the window, offering more Twido status information.</p>	
8	<p>If you start the Twido, the action must be confirmed.</p> <p>Click <b>OK</b> to proceed.</p>	
9	<p>After successfully starting the Twido, the <b>RUN</b> LED changes from yellow to green.</p>	

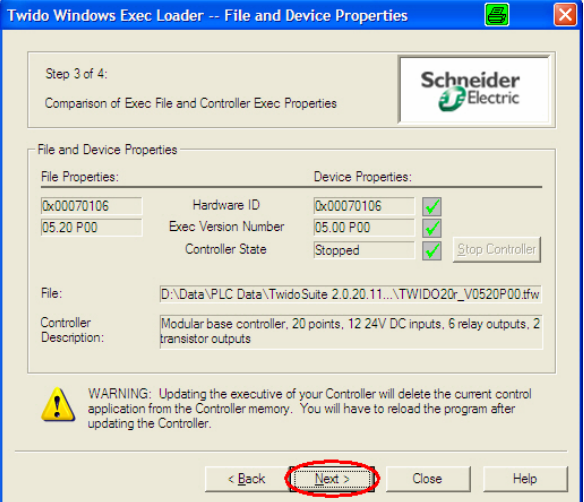

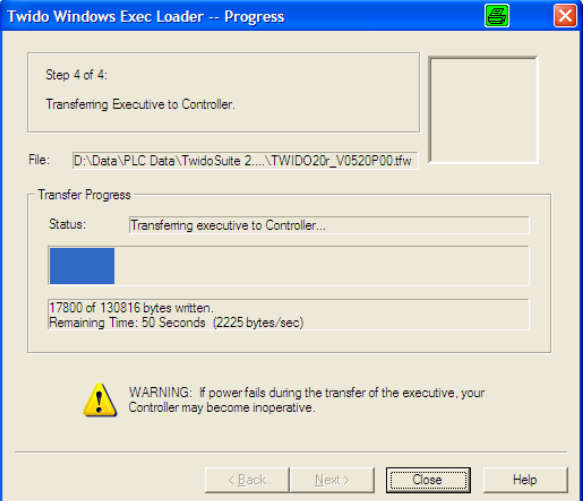
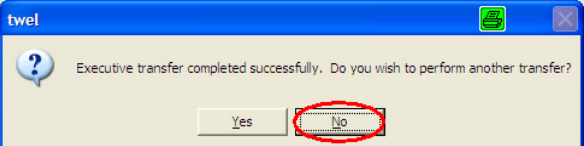
<p><b>10</b> The image on the right shows the expanded control panel with the Twido in run mode.</p>	
<p><b>11</b> If the control panel is closed, a click in the upper left hand corner of the main window re-opens it.</p>	
<p><b>12</b> The sub-menu <b>Manage animation tables</b> allows access to the animation tables.</p>	

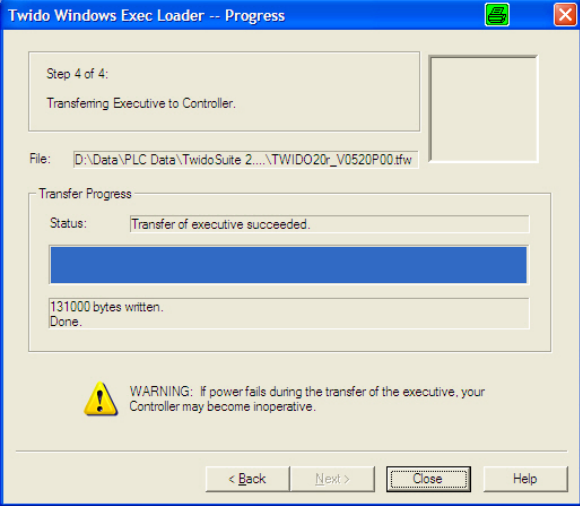
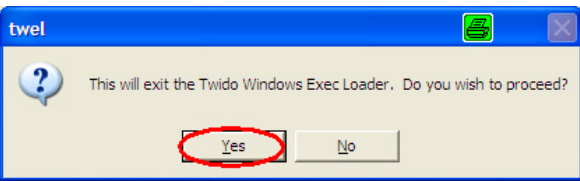
13	<p>Use the project browser to navigate through the program sections.</p>	
----	--	--

## Download firmware

1	<p>Ensure you have connected the Twido with the USB cable.</p> <p>Select:</p> <p><b>PLC firmware update</b></p>	
2	<p>You are notified that the firmware download will delete any program running on the Twido.</p> <p>Continue by clicking <b>Next &gt;</b>.</p>	

<p><b>3</b></p>	<p>Select the correct <b>Connection port</b>. In this case select <b>COM19</b>. (corresponding to the settings of the TSXCUSB485 adapter in your Windows Device Manager for the COM &amp; LPT ports)</p> <p>Select the file to be downloaded with <b>Browse...</b> button.</p> <p>Click on <b>Next &gt;</b> to continue.</p> <p><b>Note:</b> The firmware files are not part of the software delivery and needed to be downloaded separately from the Schneider Electric webpage.</p>	
<p><b>4</b></p>	<p>A comparison of the hardware and existing firmware is shown, as well as the controller state. If the controller is in the running state, it must be stopped before the firmware can be updated.</p> <p>Click on <b>Stop Controller</b>.</p>	
<p><b>5</b></p>	<p>Confirm the stop with <b>Yes</b>.</p>	

<p><b>6</b></p>	<p>The new Twido status is displayed in <b>File and Device Properties</b> (green tick in <b>Stopped</b>).</p> <p>Continue with <b>Next &gt;</b>.</p> <p><b>Note:</b> Updating the executive of your controller will delete the current control application from the controller Memory. You will have to reload the program after updating the controller.</p>	
<p><b>7</b></p>	<p>Confirm the notification not to interrupt the download with <b>Yes</b>.</p>	
<p><b>8</b></p>	<p>A progress bar is displayed.</p>	
<p><b>9</b></p>	<p>A message confirms the successful download and asks if you wish to download more.</p> <p>Click on <b>No</b> if you are finished.</p>	

10	In the progress dialog click on <b>Close</b> .	
11	Click on <b>Yes</b> to exit the loader.	

# HMI

## Introduction

In this application a Magelis HMISTU655 touch display is implemented. The display communicates via Modbus protocol with the Twido.

Configuration and programming of the display is done using the Vijeo Designer HMI software.

## Requirements

Before starting with Vijeo Designer the following preconditions must be met:

- Vijeo Designer software is installed on your PC
- The project **Optimized\_CANopen\_Twido.vdz** has been copied on the PC
- The Magelis HMI is switched on and connected to the PC with the cable BMXXCAUSBH018

## Procedure

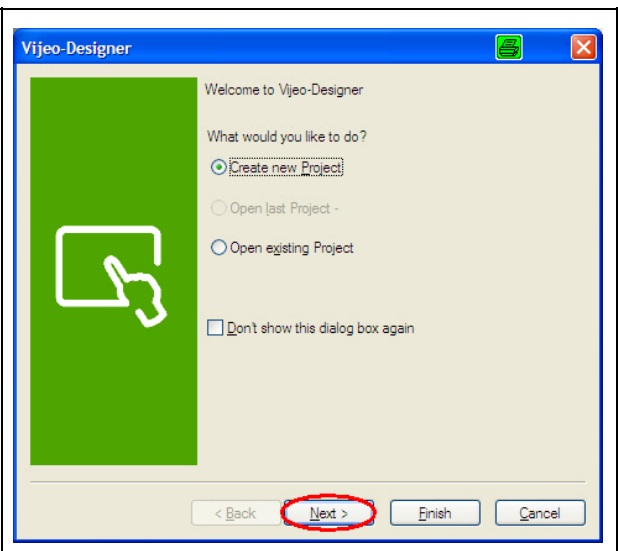
Setting up the HMI is done as follows:

- Create a new configuration
- Main window
- Configure the connection
- Create a new variable
- Create a switch
- Create a numeric display
- Create an alarm and alarm panel
- Export Vijeo Designer project
- Import Vijeo Designer project
- Download program to the HMI
- Screen navigation in the application program

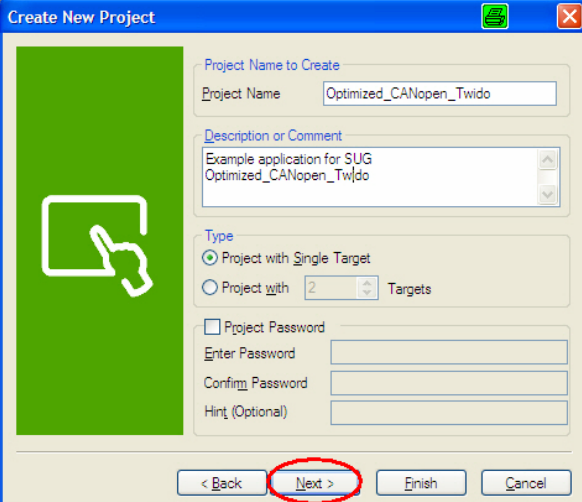
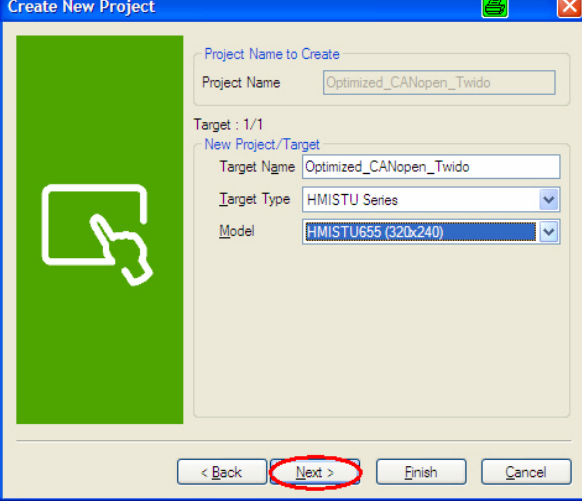
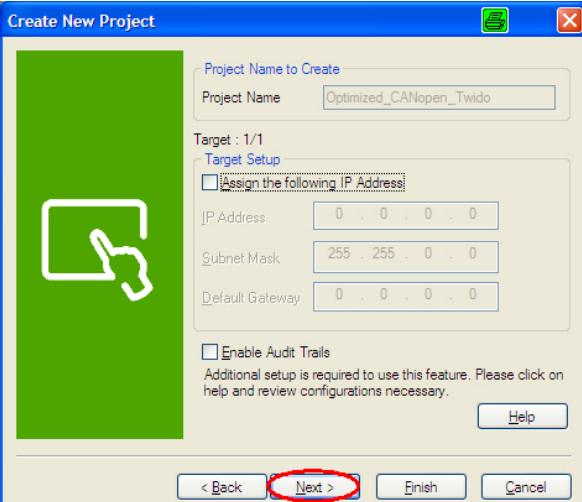
## Create a new configuration

1 When **Vijeo Designer** starts a welcome screen is opened.

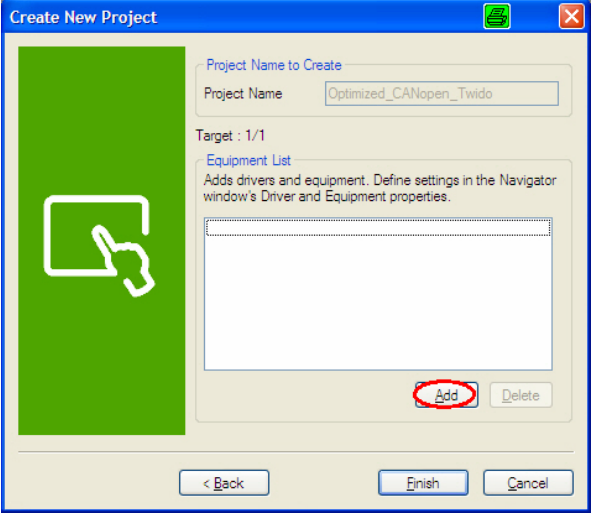
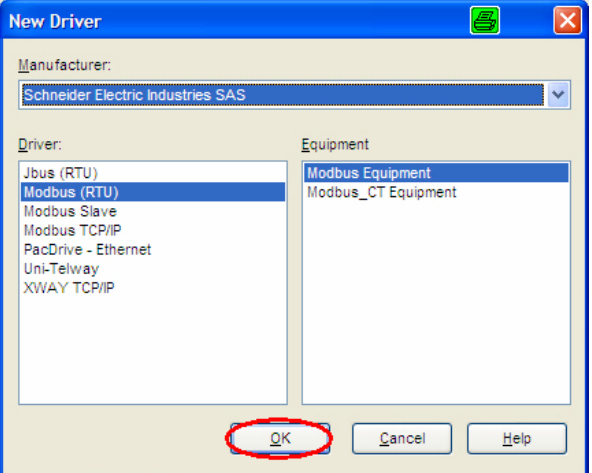
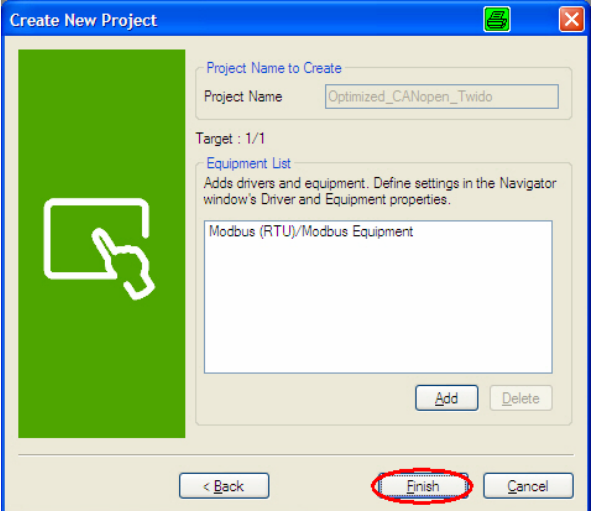
Select **Create new Project** from the list and click on **Next >**.





<p><b>2</b></p>	<p>In the field <b>Project Name</b> you need to give a name for the project, for this example</p> <p>Optimized_CANopen Twido has been entered.</p> <p>You can type in as well a description or comment.</p> <p>For this architecture the HMI is only connected to one controller so the Type of this example is <b>Project with single target</b>.</p> <p>Continue by clicking <b>Next &gt;</b>.</p>	
<p><b>3</b></p>	<p>Select the following target type and model from the drop down list:</p> <p><b>Target Name:</b> Optimized_CANopen_Twido</p> <p><b>Target Type:</b> HMISTU series</p> <p><b>Model:</b> HMISTU655 (320x240)</p> <p>Click on <b>Next &gt;</b> to continue.</p>	
<p><b>4</b></p>	<p>Leave this screen as it is without checking any box and continue by clicking <b>Next &gt;</b>.</p>	



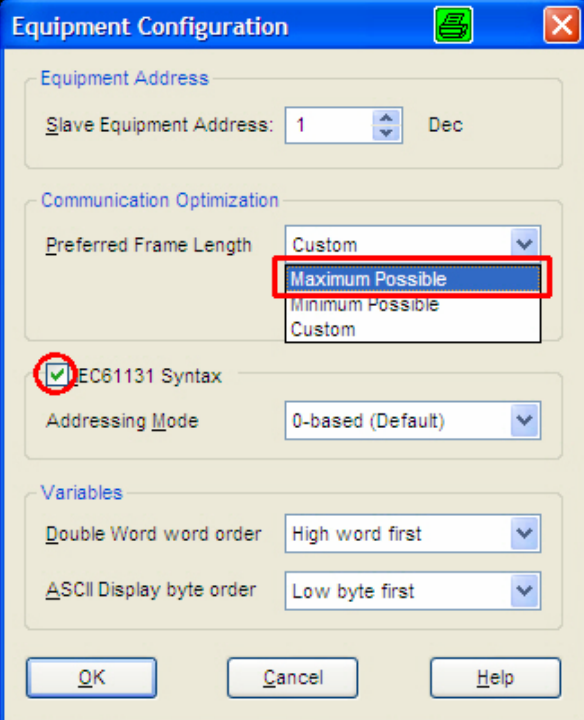
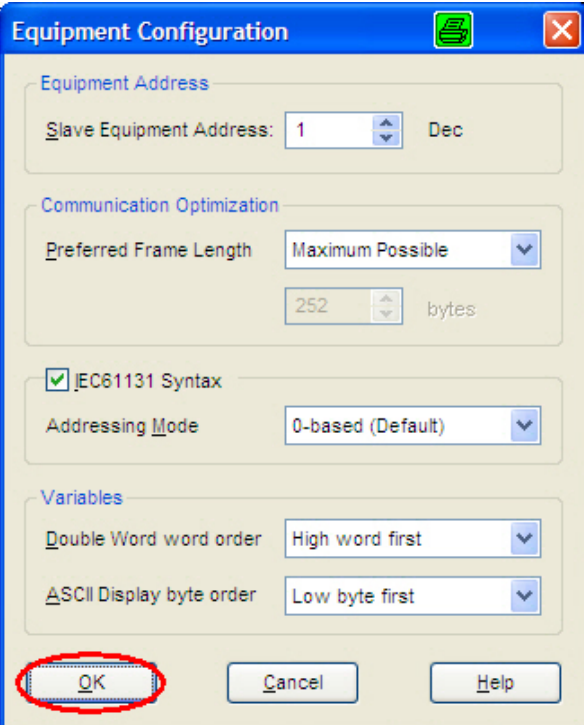
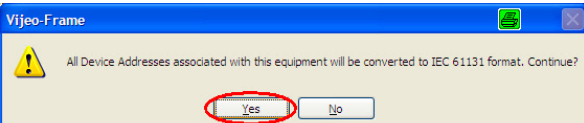
5	<p>To communicate with external controllers, additional drivers are needed. Click on <b>Add</b> to open a selection window.</p>	
6	<p>Select from the <b>New Driver</b> dialog the following settings:</p> <p><b>Manufacturer:</b></p> <ul style="list-style-type: none"> <li>- <b>Schneider Electric Industries SAS</b></li> </ul> <p><b>Driver:</b></p> <ul style="list-style-type: none"> <li>- <b>Modbus (RTU)</b></li> </ul> <p><b>Equipment:</b></p> <ul style="list-style-type: none"> <li>- <b>Modbus Equipment</b></li> </ul> <p>Close the window by clicking <b>OK</b>.</p>	
7	<p>The Modbus RTU driver is now added to the equipment list.</p> <p>To continue press <b>Finish</b>.</p>	

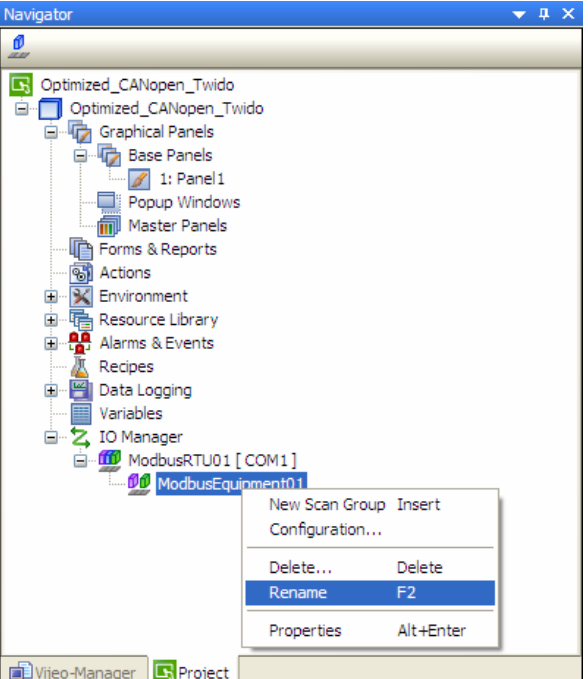
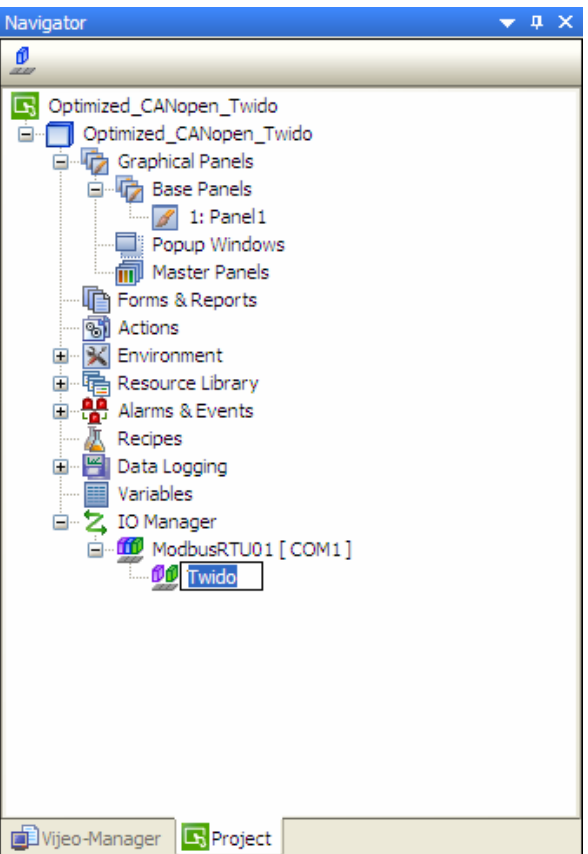
## Main Window

1	<p>After creating a Vijeo Designer HMI project the main window of Vijeo Designer is displayed.</p> <p>Vijeo Designer has the following components:</p> <ol style="list-style-type: none"> <li>1. Navigator</li> <li>2. Info viewer</li> <li>3. Toolchest</li> <li>4. Property Inspector</li> <li>5. Feedback Zone</li> <li>6. Graphic List</li> </ol>	
---	---	--

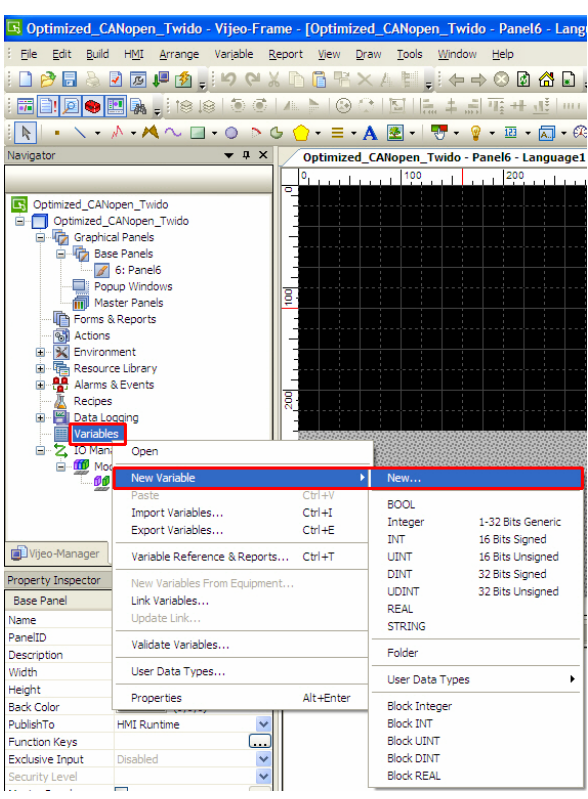
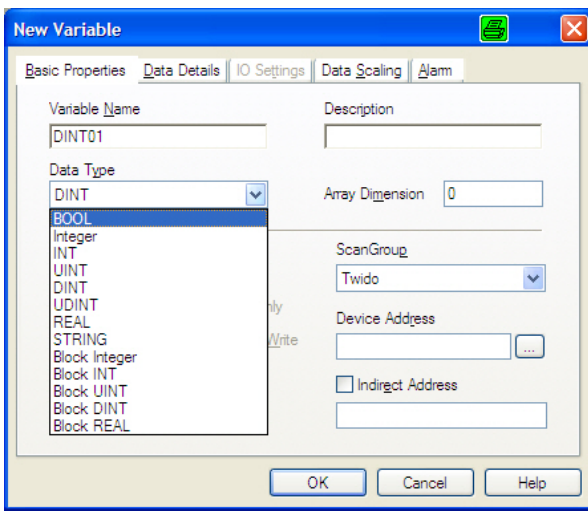
## Configure the connection

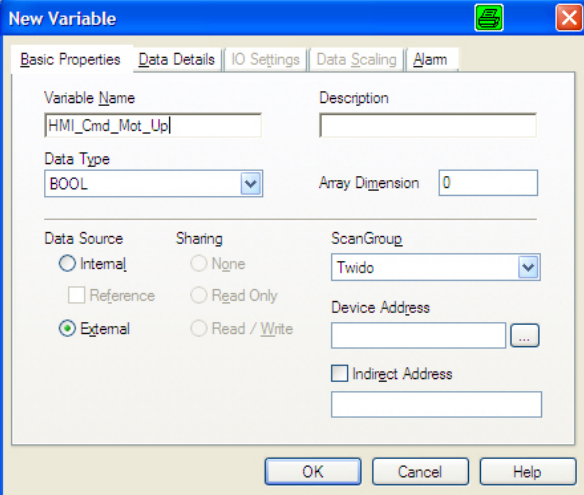
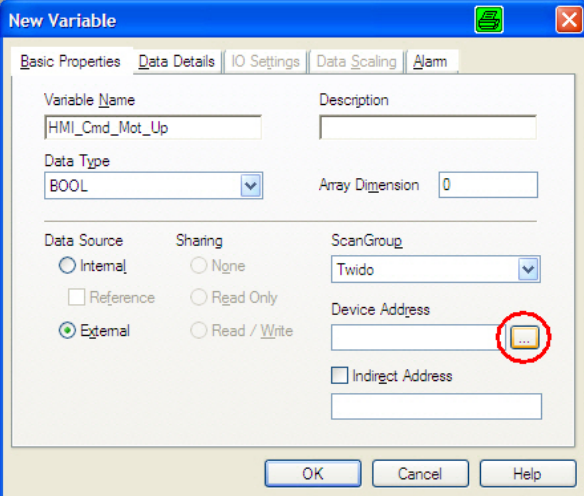
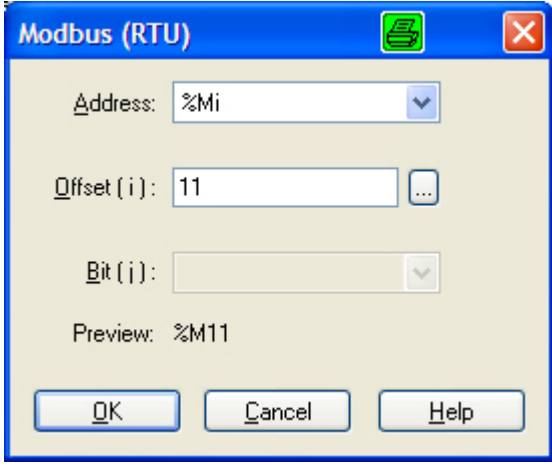
1	<p>The <b>Navigator</b> browser lists the different elements of the HMI project.</p> <p>Right click on <b>ModbusEquipment01</b>.</p> <p>Select <b>Configuration</b> in the pop-up menu.</p>	
---	---	--

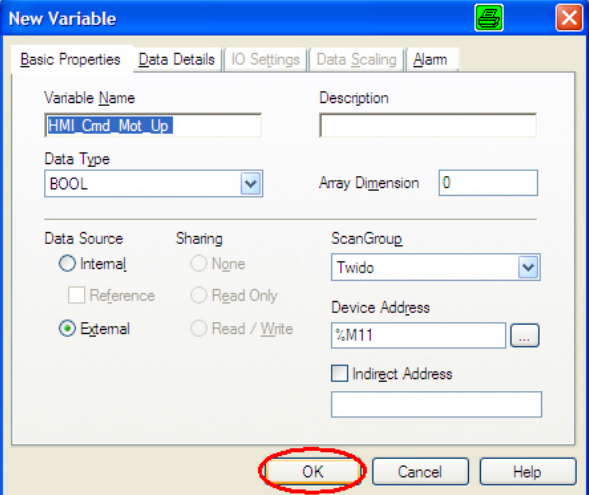
<p><b>2</b></p>	<p>The <b>Equipment Configuration</b> dialog opens.</p> <p>The <b>Slave Equipment Address</b> is already set to 1 which corresponds with the setting in Twido configuration. Following two changes need to be applied in addition:</p> <ul style="list-style-type: none"> <li>▪ Set the <b>Preferred Frame Length</b> to <b>Maximum Possible</b></li> <li>▪ Enable the checkbox for <b>IEC61131 Syntax</b></li> </ul> <p>Confirm the changes by clicking <b>OK</b>.</p>	
<p><b>3</b></p>	<p>Confirm the changes by clicking <b>OK</b>.</p>	
<p><b>4</b></p>	<p>A message pops up to inform that all device addresses with this equipment will be converted to IEC61131 format. Click <b>Yes</b> to confirm.</p>	

<p>5</p>	<p>To make the link between the <b>ModbusEquipment01</b> and the Twido more obvious you can rename the connection.</p> <p>With a right mouse click on <b>ModbusEquipment01</b> a pop-up menu opens where you select <b>Rename</b>.</p>	 <p>The screenshot shows the 'Navigator' window with a tree view of the project. Under 'IO Manager', 'ModbusRTU01 [COM1]' is expanded, and 'ModbusEquipment01' is selected. A right-click context menu is displayed with options: 'New Scan Group', 'Insert Configuration...', 'Delete...', 'Delete', 'Rename' (highlighted), 'F2', 'Properties', and 'Alt+Enter'.</p>
<p>6</p>	<p>Type in the new name, in this example <b>Twido</b>.</p>	 <p>The screenshot shows the same 'Navigator' window, but the connection under 'IO Manager' is now named 'Twido'.</p>

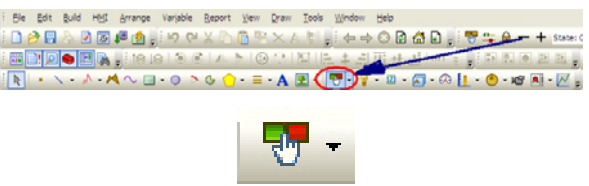
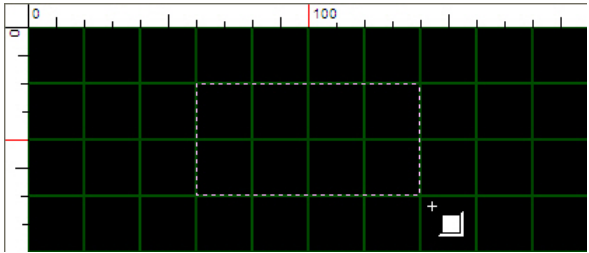
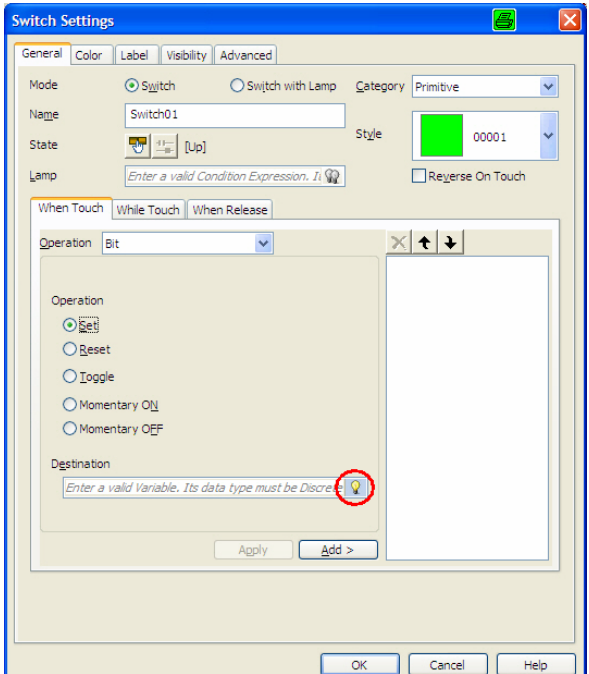
Create a new variable

1	<p>Right click on <b>Variable</b>.</p> <p>Select from the pop-up menu <b>New Variable</b> → <b>New...</b></p>	
2	<p>The <b>New Variable</b> dialog opens.</p> <p>Change the <b>Data Type</b> of the variable you wish to create by selecting the desired data type from the dropdown list.</p> <p>In this example <b>BOOL</b> is selected from the list.</p>	

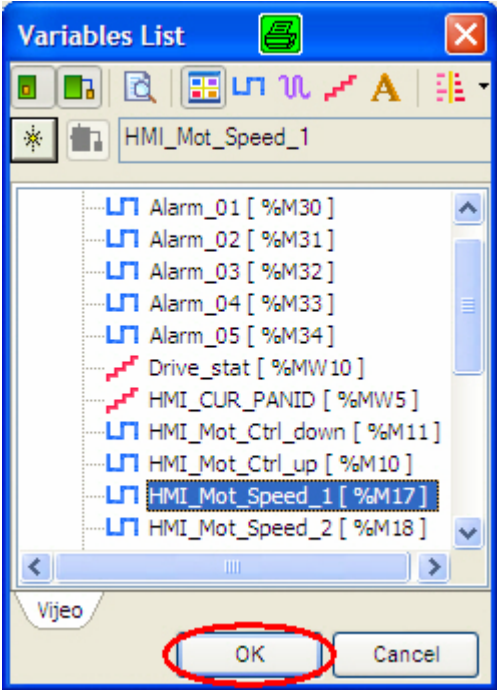
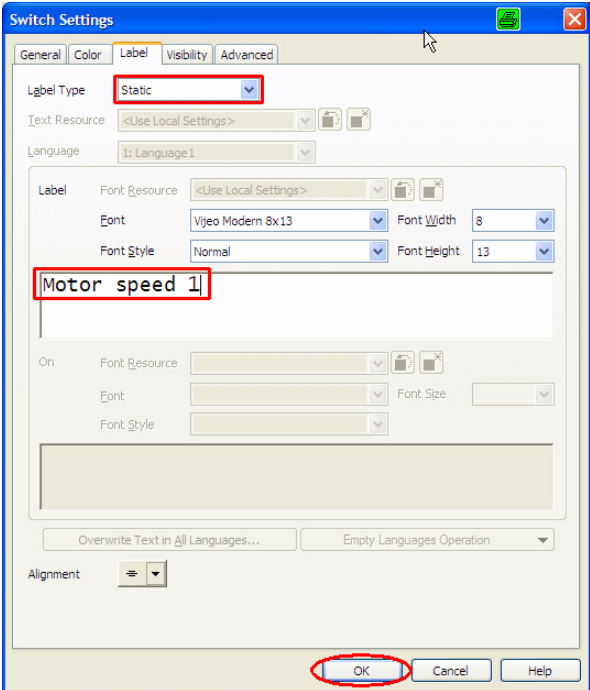
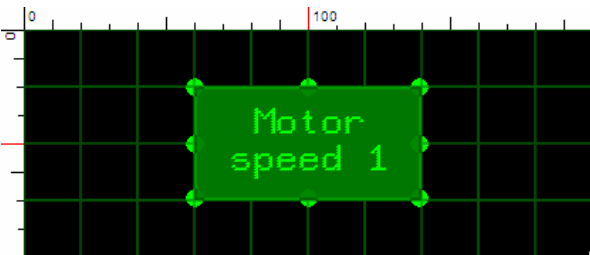
<p><b>3</b></p>	<p>Type in the <b>Variable Name</b>, in this example <b>HMI_Cmd_Mot_Up</b>.</p> <p>You can enter as well a comment or explanation to this new variable in the <b>Description</b> field.</p>	
<p><b>4</b></p>	<p>In order to communicate with the Twido the variables must be created in the same <b>ScanGroup</b> (here: <b>Twido</b>) which is linked to the previous steps <b>Configure the connection</b>.</p> <p>Click on the three dots button [...] right to the <b>Device Address</b> field.</p>	
<p><b>5</b></p>	<p>The <b>Modbus (RTU)</b> pop-up opens which allows you to locate the address of the variable.</p> <p>In this example the address <b>%M11</b> will be created.</p> <p>Select from the dropdown list of the <b>Address</b> field the item <b>%Mi</b> and enter <b>11</b> for the <b>Offset (i)</b>.</p> <p>The address <b>%M11</b> gets displayed in the <b>Preview</b>.</p> <p>Confirm the changes by clicking <b>OK</b> and the pop-up menu closes.</p>	

<p><b>6</b></p>	<p>To enter more variables repeat the steps 2 to 5.</p> <p>If you have finished entering new variables, click <b>OK</b> to close the dialog box.</p>	
-----------------	--	--

## Create a switch

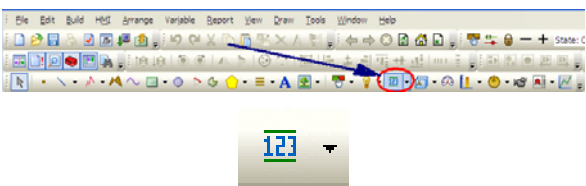
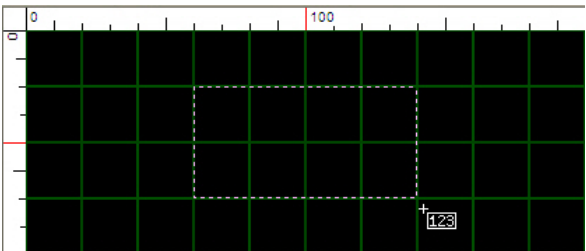
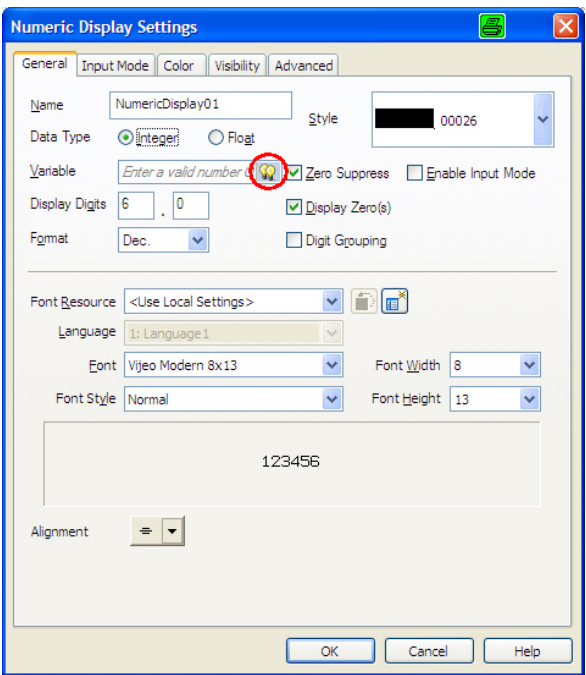
<p><b>1</b></p>	<p>Click on the <b>Switch</b> icon in the toolbar.</p>	
<p><b>2</b></p>	<p>Select the position where you want to position the switch by opening the rectangle and pressing <b>Enter</b>.</p>	
<p><b>3</b></p>	<p>The <b>Switch Settings</b> dialog opens automatically in its <b>General</b> tab.</p> <p>To link the variable to the switch click on the bulb icon to browse for a variable.</p>	

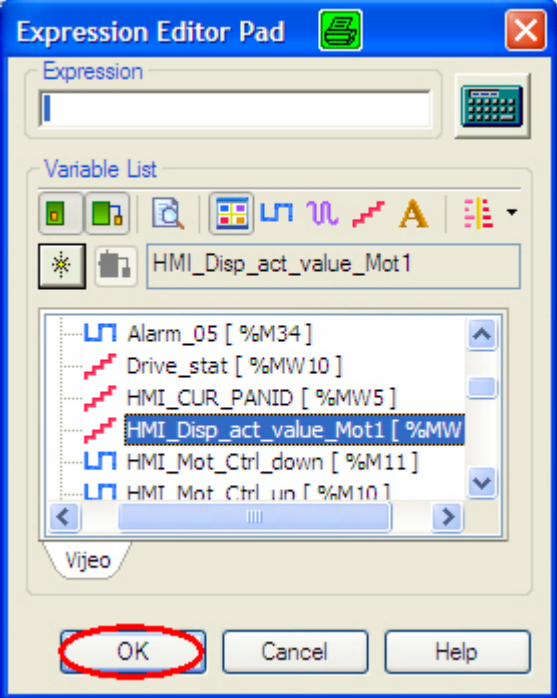
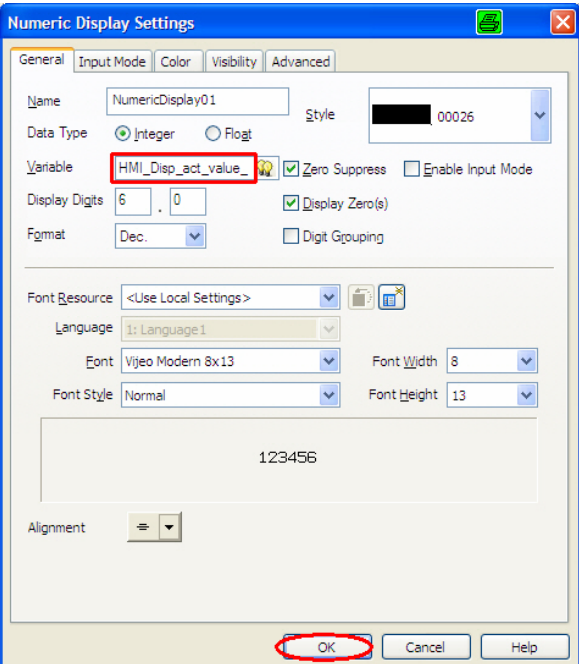
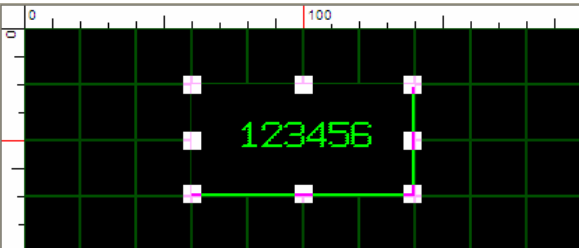


<p>4</p>	<p>Clicking on the bulb icon (as indicated in the image above) opens the <b>Variables</b> list.</p> <p>Select as variable from the list and click <b>Enter</b>.</p>	
<p>5</p>	<p>Back in the <b>Switch Settings</b> dialog, go to the Label tab.</p> <p>Here select <b>Label Type</b>.</p> <p>In the middle of this dialog you can type in the label for the switch.</p> <p>Close the dialog box by clicking <b>OK</b>.</p>	
<p>6</p>	<p>The display now shows the new button with the label you gave it.</p>	

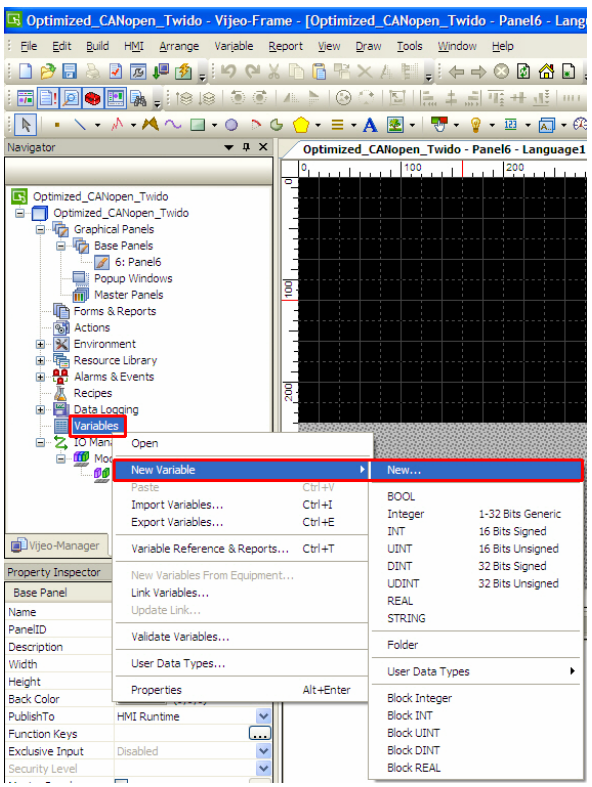
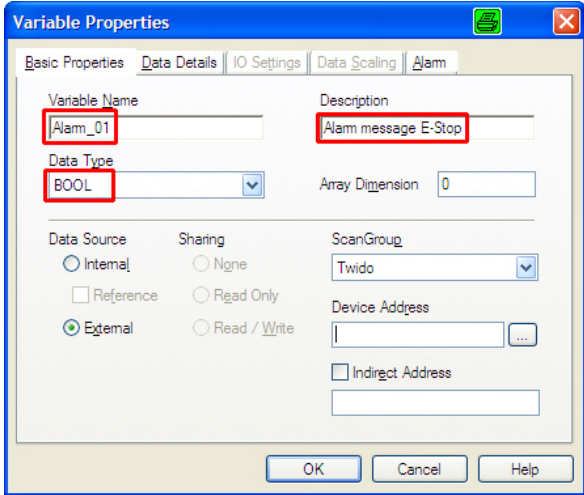


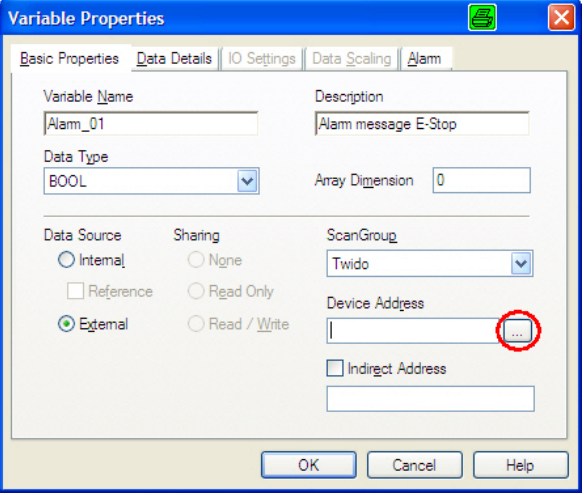
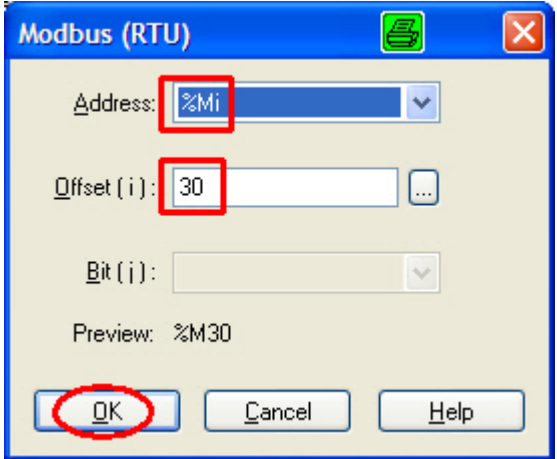
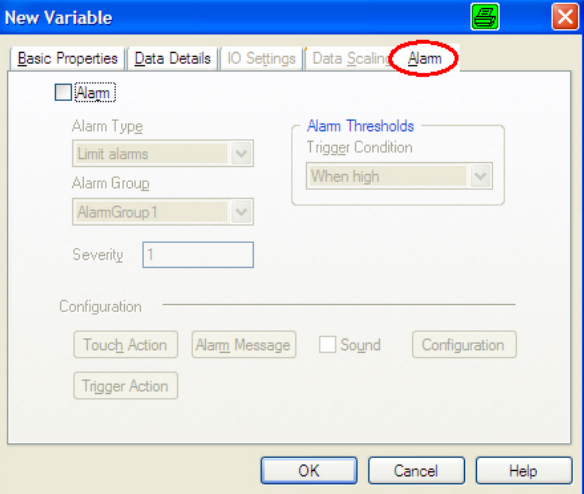
Create a numeric display

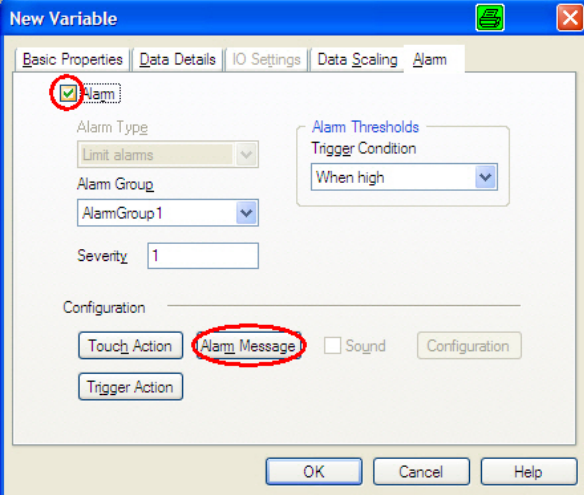
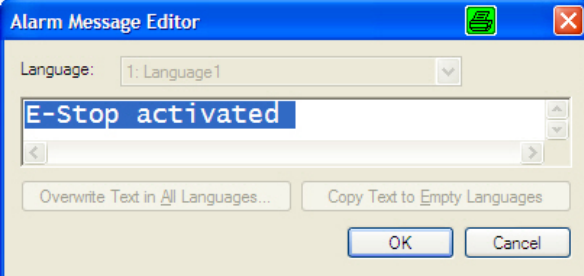
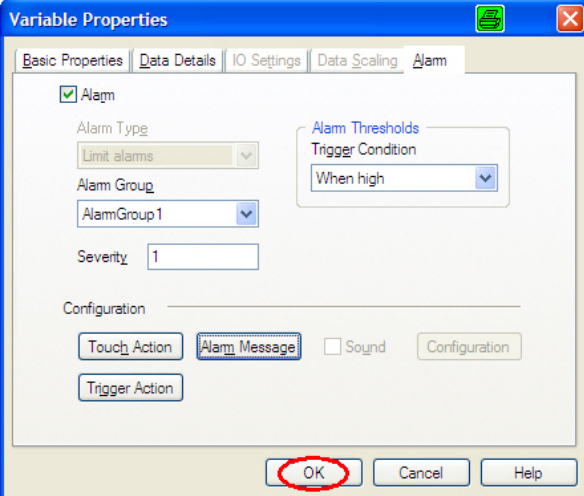
1	Click on the <b>numeric display</b> icon in the toolbar.	
2	Select the spot where you want to position the display by opening the rectangle and pressing <b>Enter</b> .	
3	<p>The <b>Numeric Display Settings</b> dialog opens automatically in its <b>General</b> tab.</p> <p>In <b>Display Digits</b> you can set the maximum number of the digits to be displayed for both integral and fractional part of the value.</p> <p>To link a variable to the display click on the <b>double bulb</b> icon to browse for a variable.</p>	

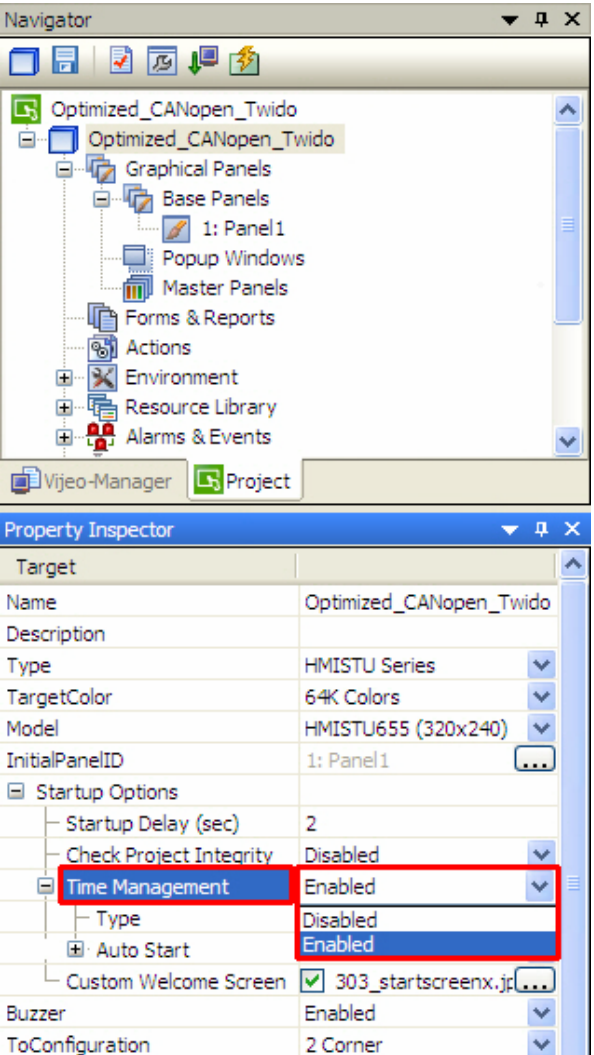
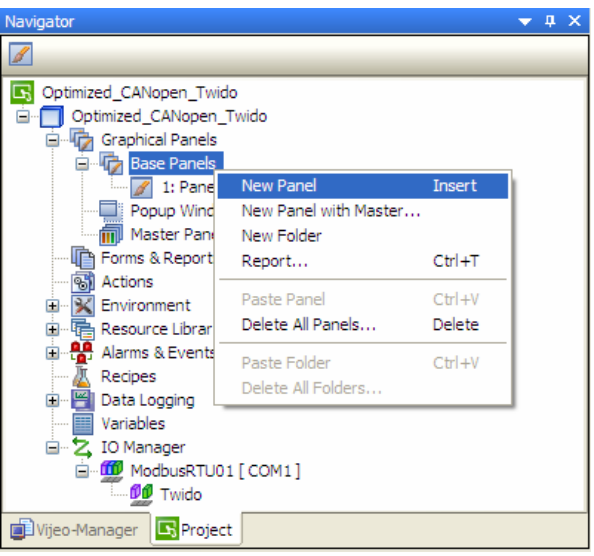
4	<p>Clicking on the double bulb icon (as indicated in the image above) opens the <b>Variables</b> list.</p> <p>Select a variable from the list and click <b>Enter</b>.</p>	
5	<p>The variable is now linked to the numeric display,</p> <p>Click <b>OK</b> to close the dialog box.</p>	
6	<p>The display shows the new numeric display with the full number of digits you defined.</p>	

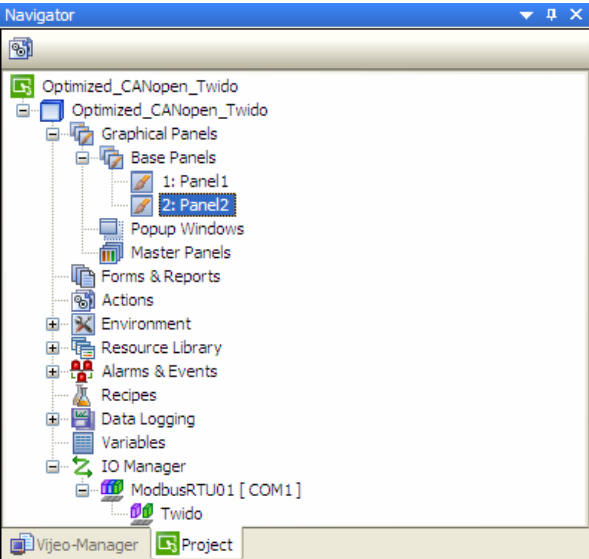
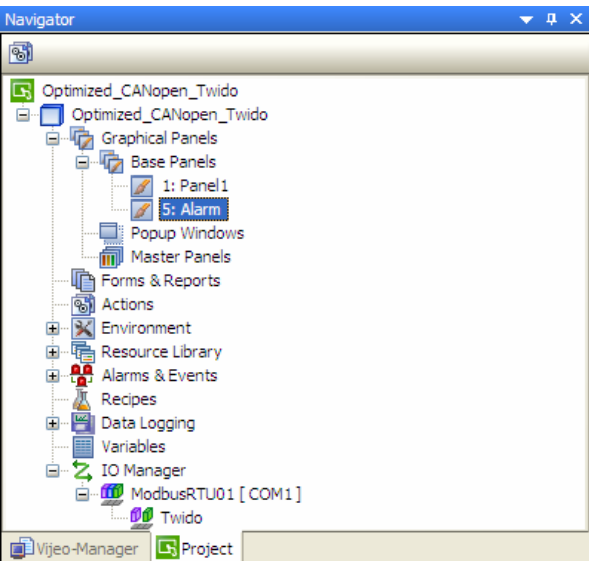
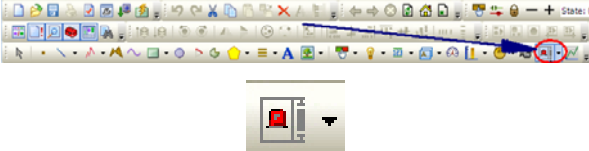
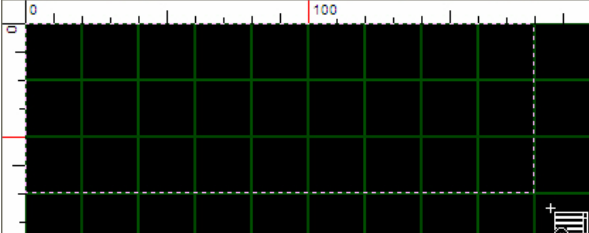
Create an alarm and alarm panel

1	<p>To create an alarm follows basically the same steps as the creation of a new variable.</p> <p>Right click on <b>Variable</b>.</p> <p>Select from the pop-up menu <b>New Variable → New...</b></p>	
2	<p>Enter in the <b>Basic Properties</b> tab Data Type, Variable Name and an optional description.</p> <p>In this example:</p> <p><b>Data Type:</b> <b>BOOL</b></p> <p><b>Variable Name:</b> <b>Alarm_01</b></p> <p><b>Description:</b> <b>Alarm message E-Stop</b></p>	

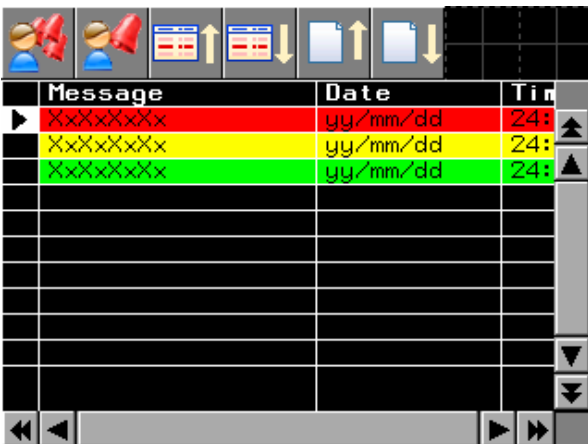
3	<p>Click on the three dots button [...] right to the <b>Device Address</b> field.</p>	 <p>The screenshot shows the 'Variable Properties' dialog box with the 'Alarm' tab selected. The 'Device Address' field is empty, and a red circle highlights the three-dot button to its right. Other fields include 'Variable Name' (Alarm_01), 'Data Type' (BOOL), and 'ScanGroup' (Twido).</p>
4	<p>Enter now the settings for variable %Mi30.</p> <p>To accept the changes and to leave the <b>Modbus (RTU)</b> pop-up, click <b>OK</b>.</p>	 <p>The screenshot shows the 'Modbus (RTU)' dialog box. The 'Address' field contains '%Mi', the 'Offset (i)' field contains '30', and the 'OK' button is circled in red. The 'Preview' field shows '%M30'.</p>
5	<p>Now switch to the <b>Alarm</b> tab.</p>	 <p>The screenshot shows the 'New Variable' dialog box with the 'Alarm' tab selected. The 'Alarm' checkbox is checked. The 'Alarm Type' is 'Limit alarms', 'Alarm Group' is 'AlarmGroup1', and 'Severity' is '1'. The 'Alarm Thresholds' section shows 'Trigger Condition' as 'When high'. The 'OK' button is circled in red.</p>

<p><b>6</b></p>	<p>Mark the <b>Alarm</b> checkbox to enable the alarm handling properties for this variable.</p> <p>By default the alarm will be triggered when the status of the variable is set to status high.</p> <p>Automatically this alarm is linked to the <b>Alarmgroup1</b>.</p> <p>Click on <b>Alarm Message</b> to associate a text which will be displayed when an alarm occurs.</p>	
<p><b>7</b></p>	<p>The <b>Alarm Message Editor</b> pops up, Enter in the field the text for the message which will be displayed in case the variable %M30 is active.</p> <p>Confirm and store the changes by clicking <b>OK</b>. The dialog will be closed.</p>	
<p><b>8</b></p>	<p>The new alarm has now been created. Close this dialog by clicking <b>OK</b>.</p>	

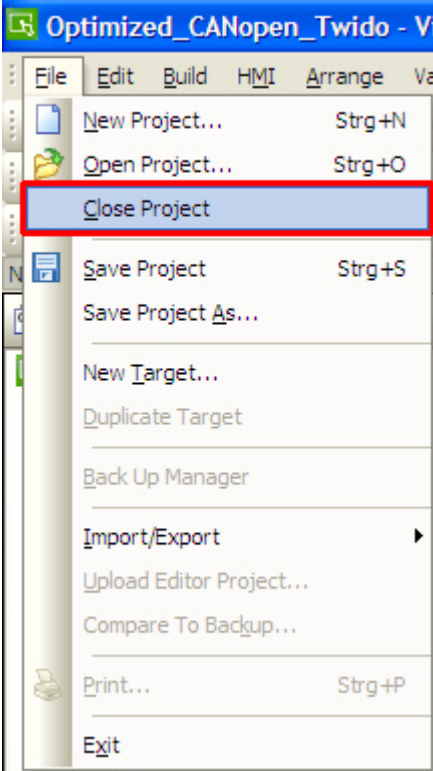
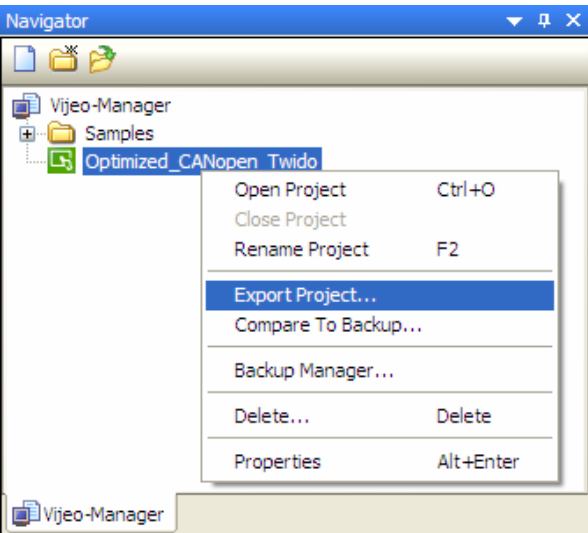
<p><b>9</b></p>	<p>The internal alarm management of the HMI will display the date and time whenever this alarm occurs.</p> <p>By default the time management capabilities of the panel are switched off.</p> <p>To activate this function click in the Navigator area on the target name and change in the Property Inspector the <b>Time Management</b> (underneath the Startup options) to <b>Enabled</b>.</p>	 <p>The screenshot shows the HMI configuration software interface. The 'Navigator' pane on the left displays a tree structure of the project. The 'Optimized_CANopen_Twido' target is selected. The 'Property Inspector' pane on the right shows the properties of the selected target. The 'Time Management' property is highlighted with a red box and set to 'Enabled'.</p>
<p><b>10</b></p>	<p>To create an alarm page in the HMI create first a new application screen.</p> <p>With a right mouse click on <b>Base Panels</b> a pop-up opens where you select <b>New Panel</b>.</p>	 <p>The screenshot shows the HMI configuration software interface. The 'Navigator' pane on the left displays a tree structure of the project. The 'Base Panels' folder is selected, and a right-click context menu is open. The 'New Panel' option is highlighted.</p>

11	<p>A new panel is created with the next free ID and a default name.</p>	
12	<p>You can change the panel ID and as well its name by easily typing in.</p> <p>In this example it is: <b>5: Alarm</b></p>	
13	<p>Click on the <b>Alarm Summary</b> icon in the tool bar.</p>	
14	<p>Select the position where you wish to place the alarm summary by opening a rectangle on the display.</p>	

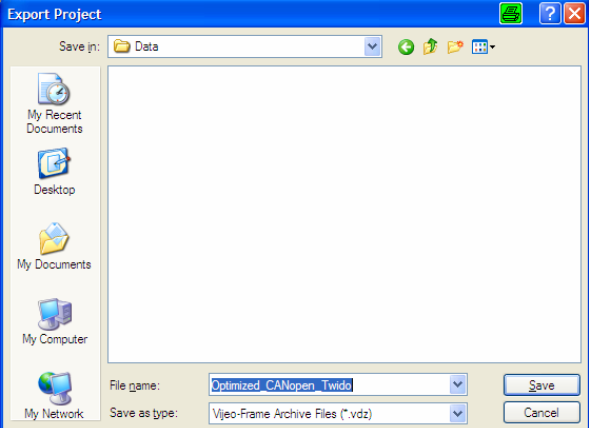
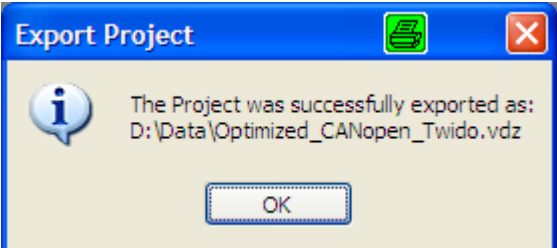


15	The Alarm page is created on its own panel.	
----	---	--

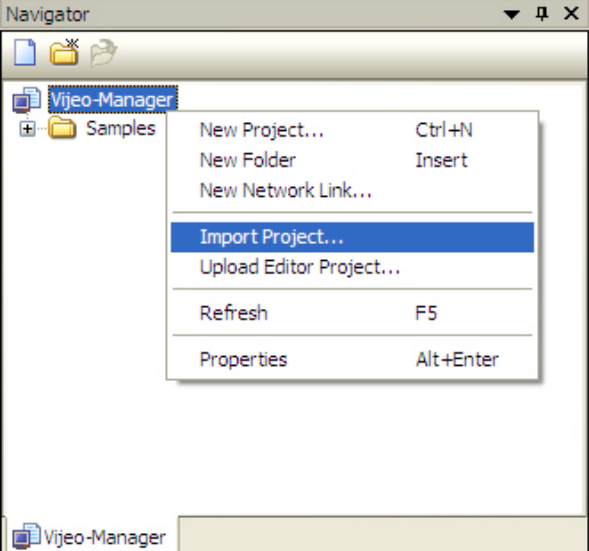
## Export Vijeo Designer project

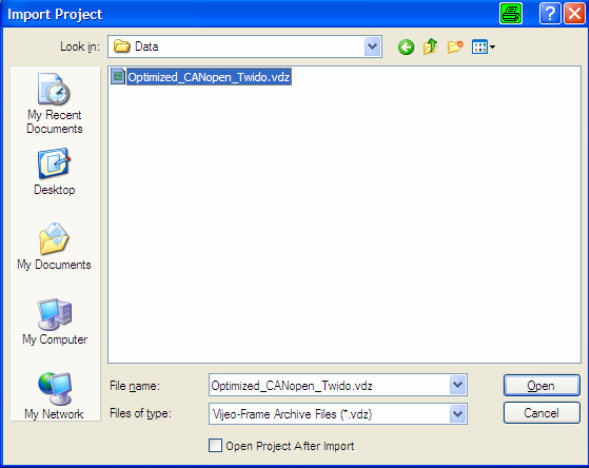
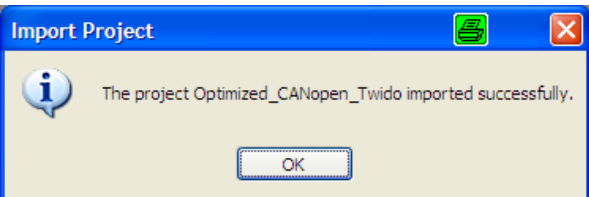
1	<p>To transfer a project from one PC to another an export of the project is required.</p> <p>First of all any open project needs to be closed. Select from the menu <b>File → Close Project</b>.</p>	
2	<p>Right click on the project you wish to export.</p> <p>Select from the pop-up menu <b>Export Project...</b></p>	



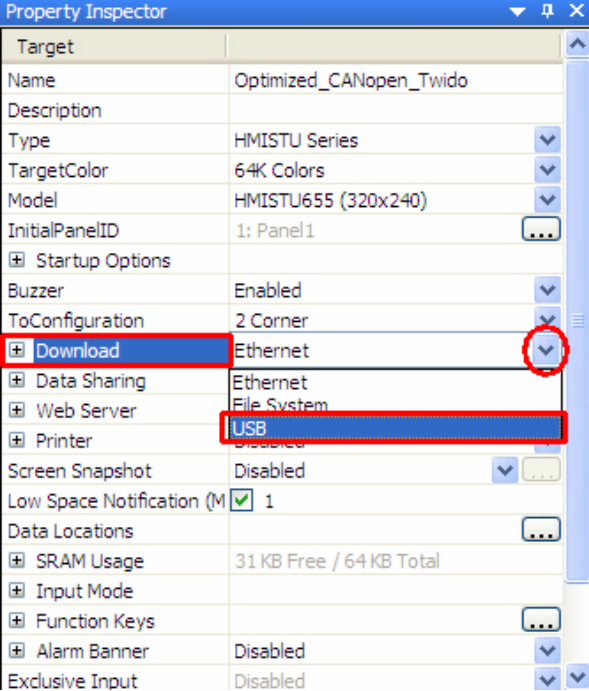
<p><b>3</b></p>	<p>A window opens where you can browse for the location the export file should be stored in.</p> <p>The default name of the export file has the same name you have given to the project. In this example it is <b>Optimized_HW_Twido</b>.</p> <p>Click on <b>Save</b> to continue the export.</p>	
<p><b>4</b></p>	<p>A message window pops up to inform you that the creation of the export file <b>Optimized_HW_Twido.vdz</b> was successful.</p>	

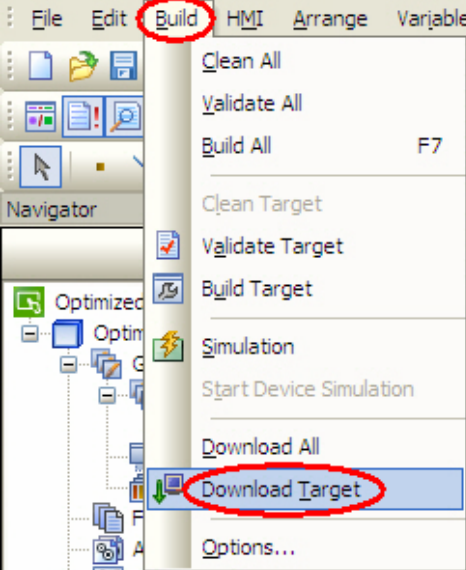
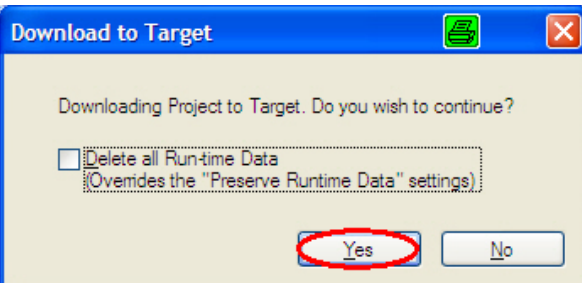
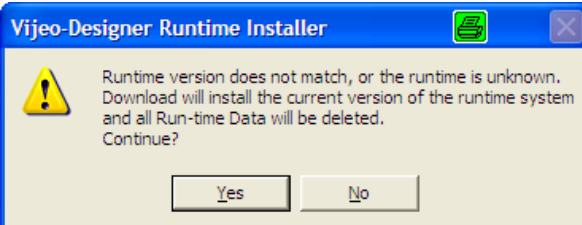
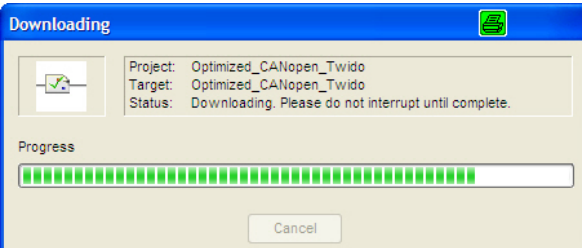
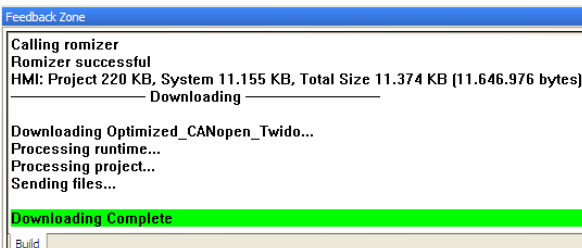
## Import Vjeo Designer project

<p><b>1</b></p>	<p>To import an existing project into Vjeo Designer, right click on <b>Vjeo-Manager</b> in the Navigator area.</p> <p>Select <b>Import Project...</b> from the pop-up menu.</p> <p><b>Note:</b> Importing a project is not possible if any project is still open.</p>	
-----------------	---	--



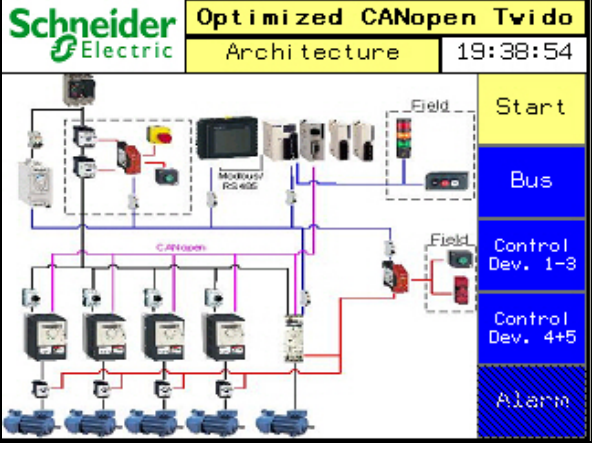
2	<p>A window opens where you can browse to the location the import file is stored in.</p> <p>Select the desired file, in this example <b>Optimized_HW_Twido.vdz</b></p> <p>Click on <b>Open</b> to continue the import.</p>	
3	<p>A message window pops up to inform you that the import of the file <b>Optimized_HW_Twido.vdz</b> was successful.</p>	

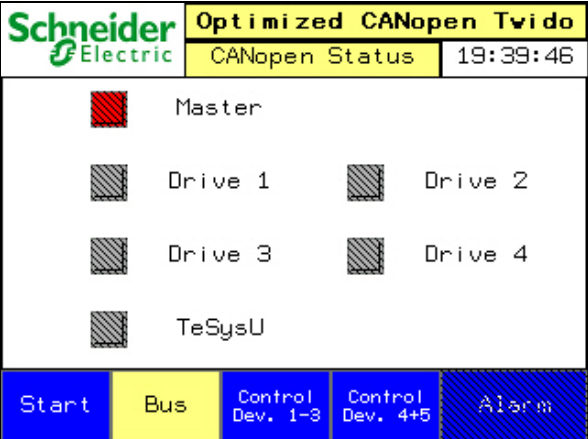
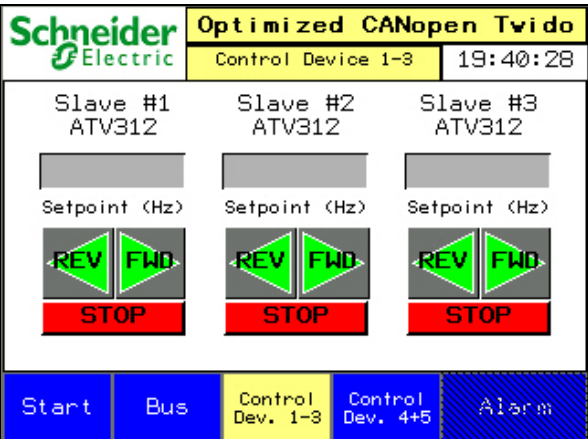
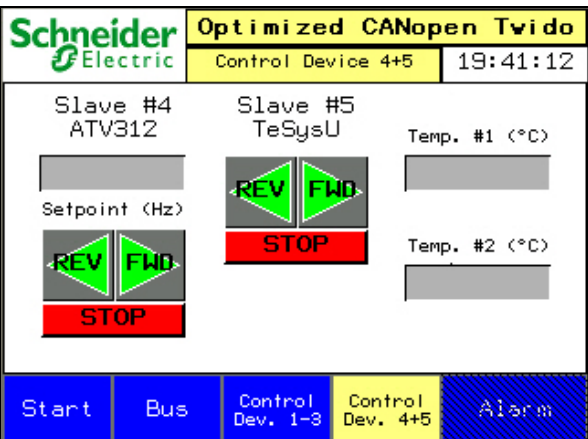
## Download Program to the HMI

1	<p>To be able to download the project to the HMI the correct download communication must be set.</p> <p>Go in the <b>Property Inspector</b> to the <b>Download</b> item and select from the two options <b>USB</b> from the dropdown list.</p>	
---	--	---

2	<p>Click on <b>Build</b> in the Menu bar and select the item <b>Download Target</b>.</p>	
3	<p>Vijeo Designer connects now to the HMI and begins the build and download process. A dialog box appears and requires your attention.</p> <p>To continue click <b>Yes</b>.</p>	
4	<p>A dialog box pops up in case the runtime version of the connected HMI does not match. Click on <b>Yes</b> to continue and the latest runtime version will be installed on the device.</p>	
5	<p>The progress is shown in the <b>Downloading</b> message box.</p>	
6	<p>After completing the download you receive a notification in the <b>Feedback Zone</b> of Vijeo Designer.</p>	

Screen navigation in the application program

1	After a power up or a restart of the HMISTU655 device the screen on the right will appear.	
2	You can click on the <b>Date/Time</b> button to edit the date and time of the HMISTO511 device. If no action is taken the screen switches after 10 seconds to the start panel.	
3	<p>The <b>Start</b> panel shows the architecture and offers on the right hand side five buttons for the available pages inside the HMI.</p> <p>The currently activated panel highlights the respective button by a yellow background.</p>	

4	<p>The <b>Bus</b> panel shows an overview of the CANopen status. It shows the master at its top and lists underneath the five CANopen slaves.</p> <p>Status <b>green</b> → Device or master not operational</p> <p>Status <b>red</b> → Device or master not operational</p> <p>Status <b>grey</b> → No response from devices because Master is not operational</p>	
5	<p>The <b>Control Device 1-3</b> panel contains the control elements for the Altivar slaves 1, 2 and 3. On the top you select the <b>Setpoint</b> speed in Hz (on controller startup it is always 10 Hz). Below this field you can <b>start</b> and <b>stop</b> the drive in both directions with the help of the pushbuttons.</p>	
6	<p>The <b>Control Device 4+5</b> panel contains the control elements for the Altivar slaves 4 and the TeSysU.</p> <p>For the ATV you select on the top the <b>Setpoint</b> speed in Hz (on controller startup it is always 10 Hz). Below this field you can <b>start</b> and <b>stop</b> the drive in both directions with the help of the pushbuttons.</p> <p>For TeSysU you have the same control elements but without setpoint speed.</p> <p>On the right the temperatures from the Pt100 thermal probe are displayed on the panel, too.</p>	

- 7

The **Alarm** panel pops up when an alarm occurs and shows the text in the message list. As long as an alarm is active the **Alarm** button in the lower right corner turns to red.

For alarms which can be reset there is an **Alarm Reset** button in the upper right part of the panel.

Schneider Electric

Optimized CANopen Twido

Alarm table

19:41:41

ALARM RESET

Message	Time	State

Start

Bus

Control Dev. 1-3

Control Dev. 4+5

Alarm

# Devices

---

## Introduction

This chapter describes the steps required to initialize and configure the devices to attain the described system function.

---

## General

Altivar 312 drives are configured by using the local control panel on the device itself. Alternatively they can be configured by using the software SoMove Lite.

## Note

If this is not a new drive you should re-establish the factory settings. If you need instructions on how to do this, please read the drive documentation.

Be sure that the controller is in STOP state before parameterizing the drives.

---

# Altivar 312

## Introduction

The ATV312 parameters can be entered or modified via the local control panel on the front of the device itself or by using the software SoMove Lite.

The advantages of using the SoMove Lite software are:

- The data can be stored on the PC
- Documentation can be printed *and*
- The tool supports online optimization of the parameters.

## Note

If this is not a new drive you should re-establish the factory settings. If you need instructions on how to do this, please read the drive documentation.

The Jog dial that is a part of the local control panel can be used for navigation by turning it clockwise or counter-clockwise. Pressing the jog dial enables the user to make a selection or confirm information.

## Procedure

Setting up the ATV is done as follows:

- ATV312 Control Panel
- Configuring ATV312 with SoMove Lite
- Download the configuration
- Connect to device / upload the configuration

## Preparation for commissioning

To speed up the commissioning, first make a note of the motor data found on the motor type labels (see example on the right) This data includes:

- **Nominal voltage**
- **Nominal power**
- **Nominal frequency**
- **Power factor  $\cos \varphi$**

### Note:

Be sure that the controller is in STOP state before parameterizing the drives.

A photograph of a motor nameplate with technical specifications. The plate is yellow with black text. It includes IP55, Ins. cl. F, S1, SF1.00, and AMB 40°C. The main table lists voltage (V), frequency (Hz), power (kW), speed (min⁻¹), current (A), and power factor (cos φ) for various motor configurations.

V	Hz	kW	min <sup>-1</sup>	A	cos φ
220 Δ	50	0.18	1340	1.03	0.79
380 Y			1340	0.597	0.79
230 Δ	50	0.18	1360	1.05	0.74
400 Y			1360	0.605	0.74
415 Y	50	0.18	1370	0.658	0.68
440 Y	60	0.21	1650	0.604	0.76
460 Y			1665	0.600	0.72

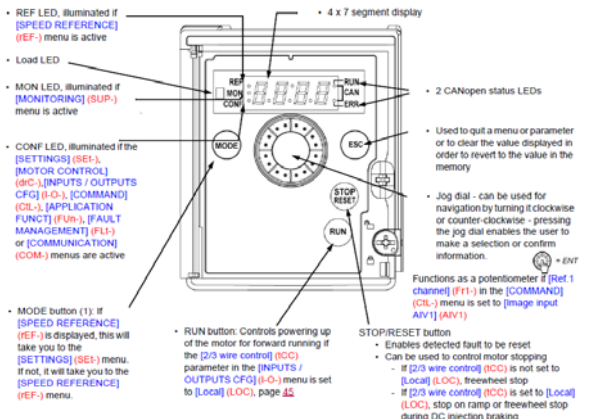


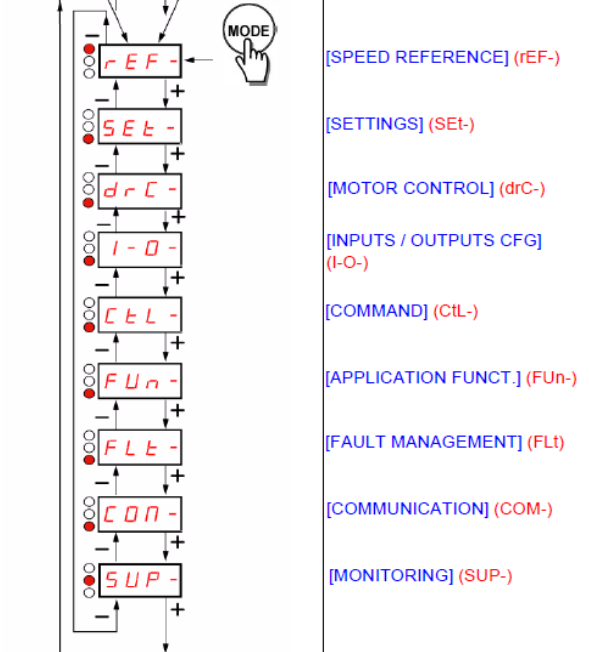
Motor name plate



## ATV312 Control Panel

The ATV312 parameters can be entered or modified via the control panel on the front of the device. This section describes how to set up the drive using the local control panel.

Alternatively, you can use the SoMove Lite software to configure the CANopen addresses and baud rate.

1	The CANopen address and Baudrate can be input using the buttons and the jog dial on the front panel of the Altivar.	
2	Using the buttons on the front panel, select the sub-menu <b>Communication</b> .	
3	In the <b>Communication</b> sub-menu input the CANopen address in the parameter <b>AdC0</b> . In the example application the addresses for the four drives are 1 to 4.	 <p>These three parameters are only visible when the drive is powered up for the first time.</p> <p>The settings can be amended subsequently in the menus:</p> <ul style="list-style-type: none"> <li>[MOTOR CONTROL] (drC-) for [Standard mot. freq] (bFr),</li> <li>[COMMAND] (CTL-) for [Ref.1 channel] (Fr1)</li> <li>[INPUTS / OUTPUTS CFG] (I-O-) for [2/3 wire control] (tCC)</li> </ul>
4	Also in the <b>Communication</b> sub-menu, in the parameter <b>BdC0</b> , set the baudrate to <b>500.0</b> (kBits).	

- |          |  |
|----------|--|
| <b>5</b> | For the ATV312 to operate with the new address or Baudrate, a power cycle (on, off, on) is required. |
|----------|--|

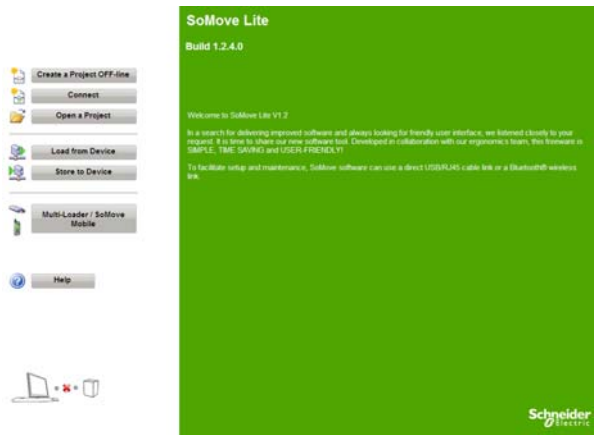
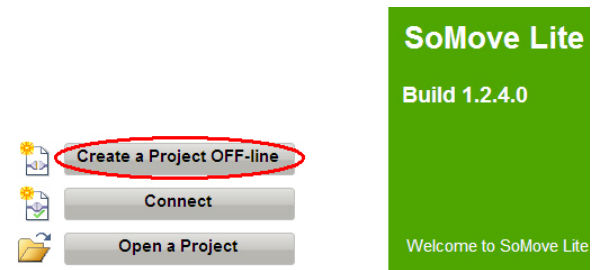
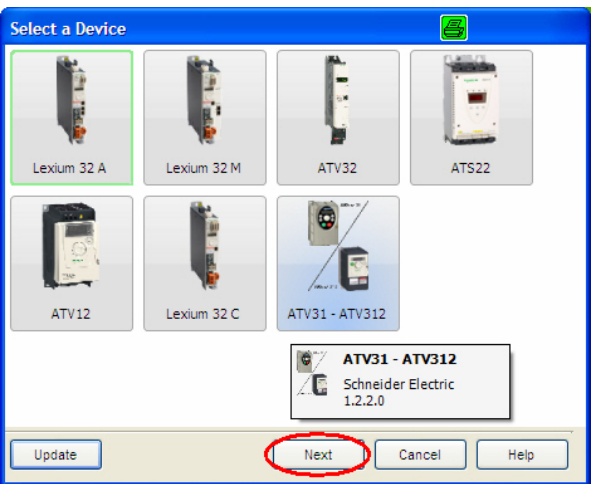
## ⚠ WARNING

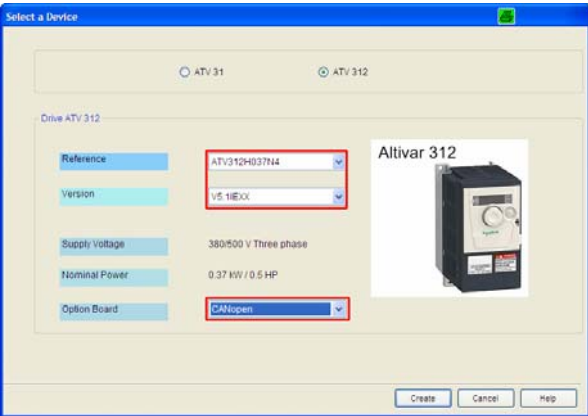
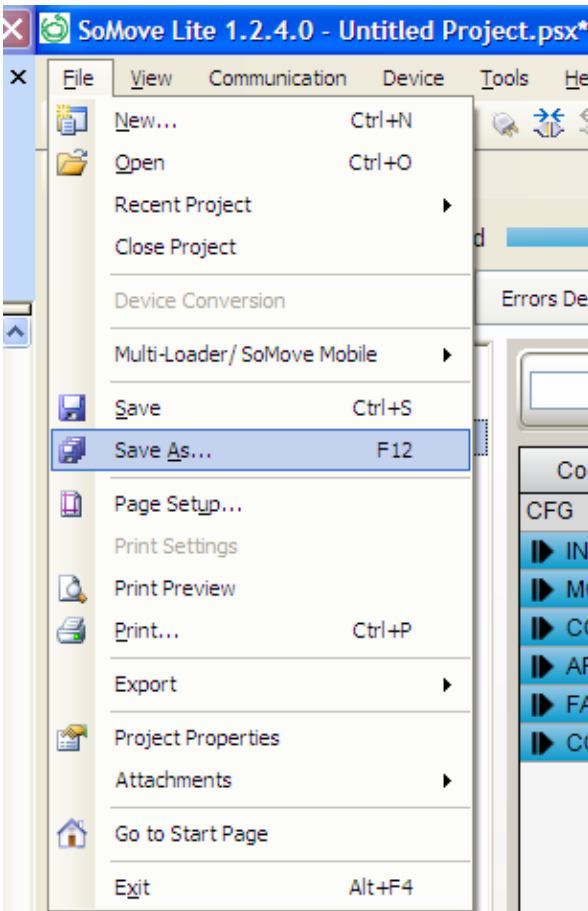
### UNINTENDED EQUIPMENT OPERATION

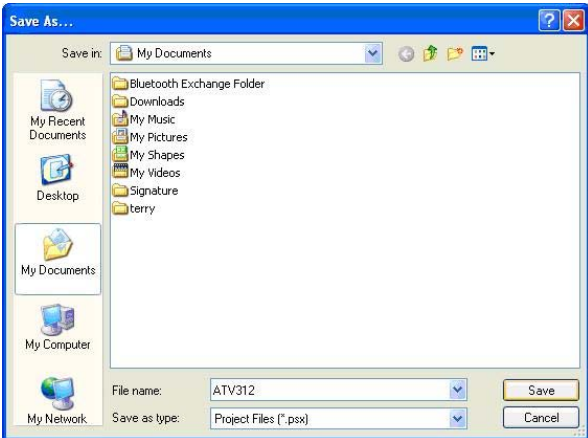
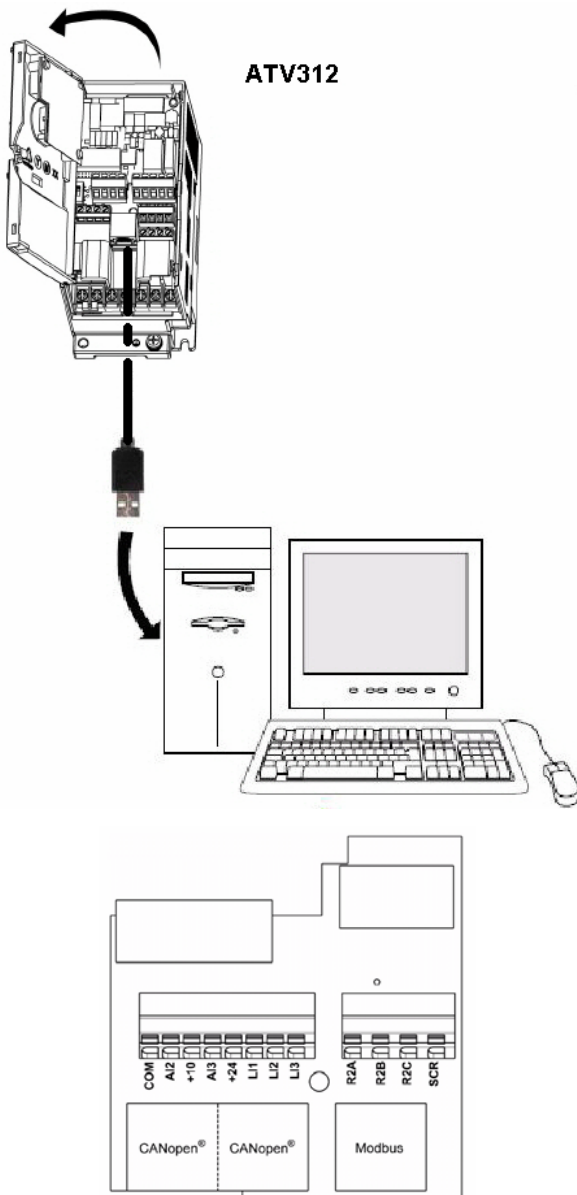
After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

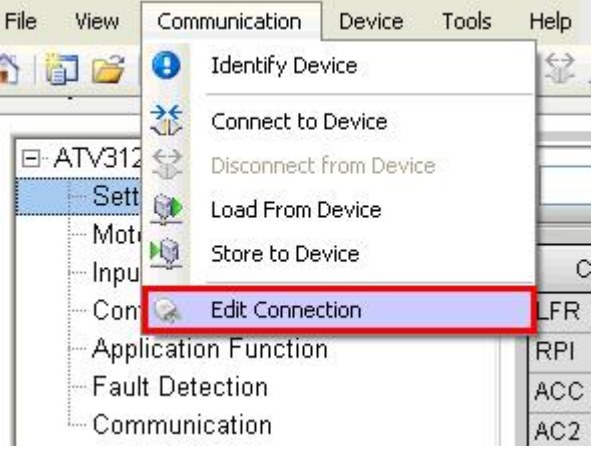
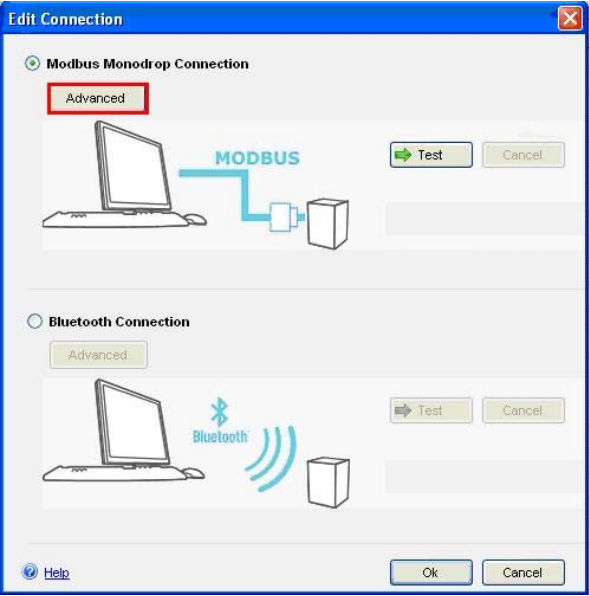
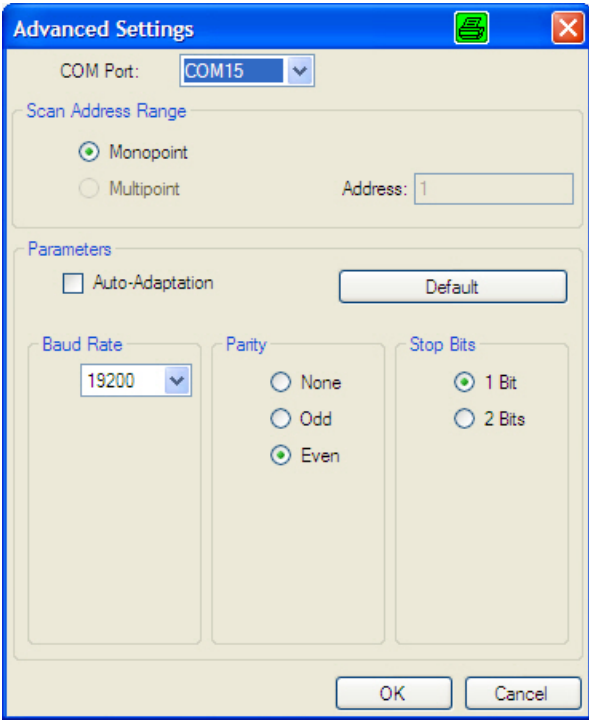
**Failure to follow these instructions can cause death, serious injury or equipment damage.**

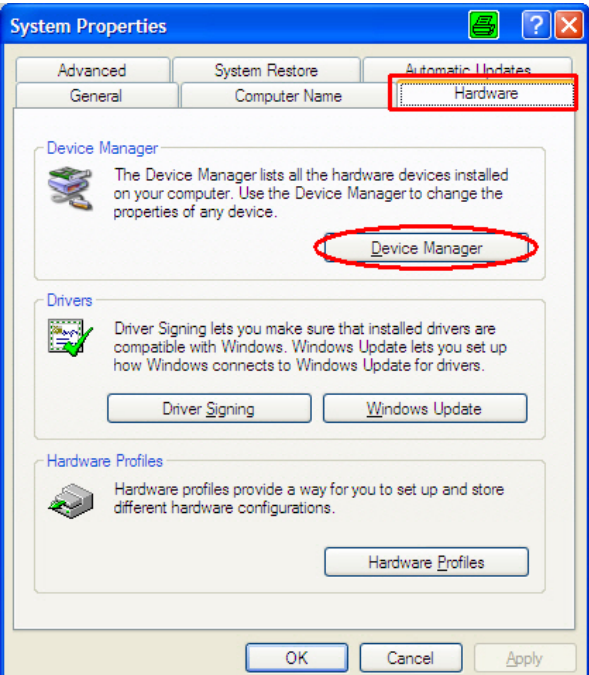
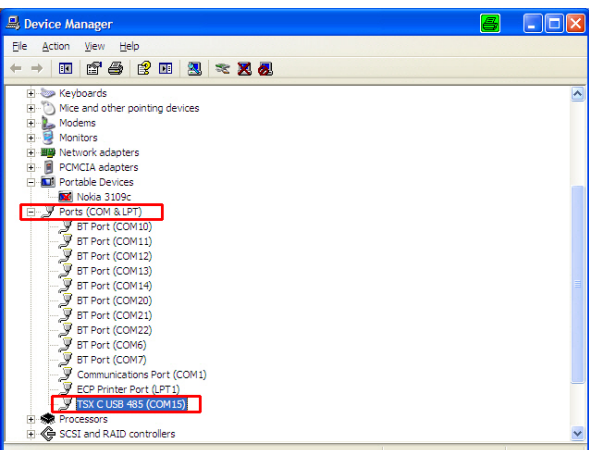
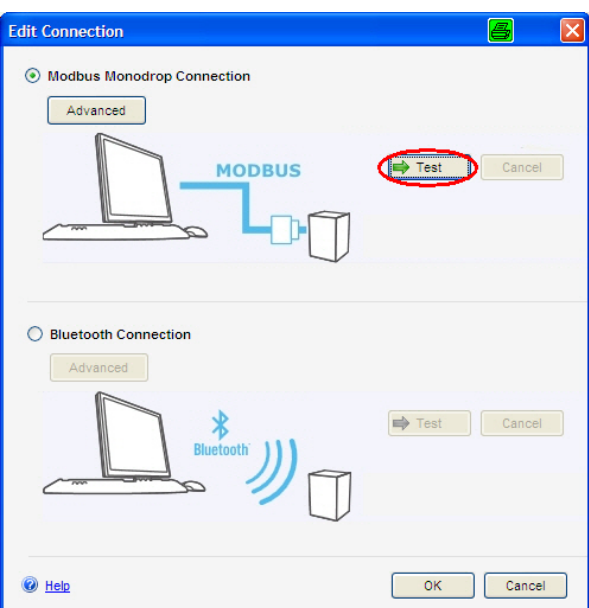
### Configuring ATV312 with SoMove Lite

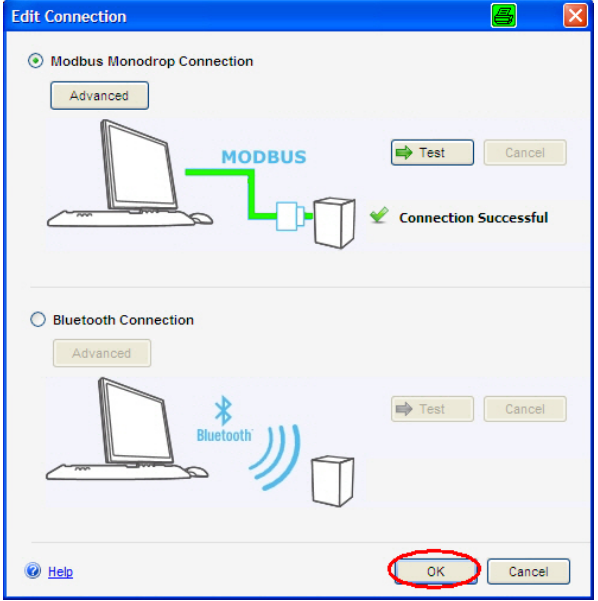
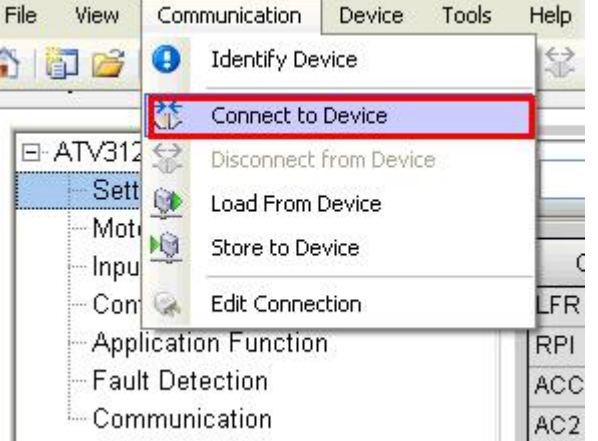
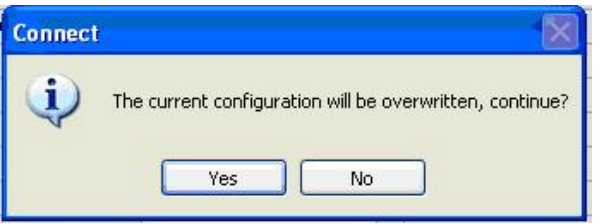
<b>1</b>	After starting SoMove Lite you will see the window as shown here on the right – the version is automatically shown in English.	
<b>2</b>	To create a new Project, select <b>Create a Project OFF-line</b> .	
<b>3</b>	Select now the desired device by clicking on the icon and continue with <b>Next</b> .	

<p>4</p>	<p>Select <b>ATV312</b> from the options. In the <b>Device</b> dialog, select a <b>Reference</b> and the <b>Firmware Version</b>.</p> <p>In this example:</p> <p><b>Reference:</b> ATV312H037N4</p> <p><b>Version:</b> V5.1IEXX</p> <p><b>Option Board:</b> CANopen</p> <p>Additional information displayed:</p> <ul style="list-style-type: none"> <li>- Supply Voltage</li> <li>- Nominal Power</li> </ul> <p>of the drive.</p> <p>Continue with <b>Create</b>.</p>	
<p>5</p>	<p>To save the project use</p> <p><b>File→Save as...</b></p>	


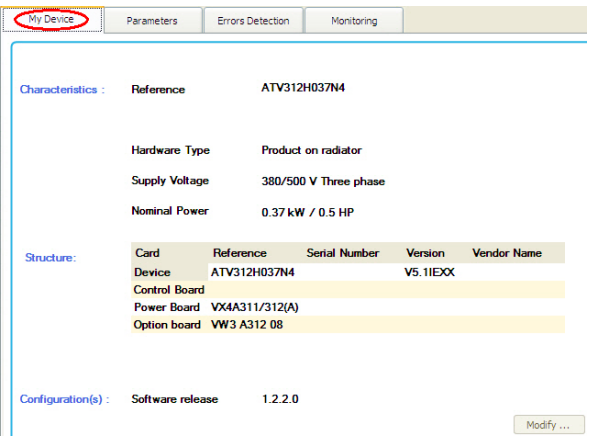
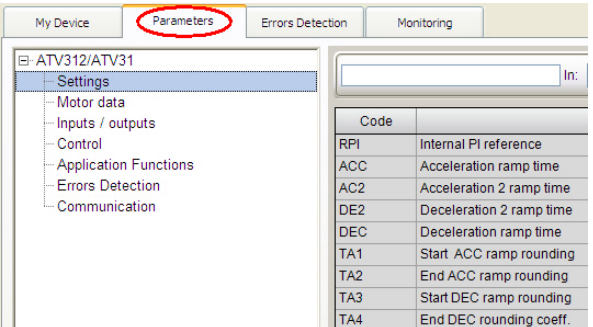
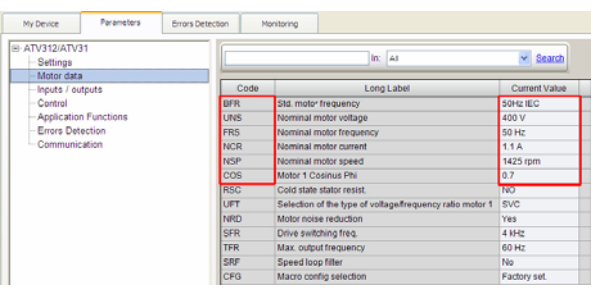
6	<p>Enter the <b>File name</b> (here ATV312) and continue with <b>Save</b>.</p>	
7	<p>Connect now the RJ45 socket on the drive with the USB port on your PC by using the programming cable <b>TCSMCNAM3M002P</b>.</p> <p><b>Note:</b> If you use the daisy chain option board for the ATV312 the RJ45 connector for Modbus is on the right.</p>	

8	<p>Select:</p> <p><b>Communication-&gt;Edit Connection</b></p> <p>to go to the connection dialog.</p>	
9	<p>Select <b>Advanced</b> to check and edit the connection settings.</p>	
10	<p>Select the appropriate <b>COM Port</b>, here <b>COM15</b> is used.</p> <p><b>Baud Rate:</b> 19200  <b>Parity:</b> Even  <b>Stop Bits:</b> 1</p> <p>Continue with <b>OK</b>.</p> <p><b>Note:</b>  If you don't know which COM port is assigned to the TCSMCNAM3M002P USB cable continue with step 11. If you already know the COM port go directly to step 13.</p>	

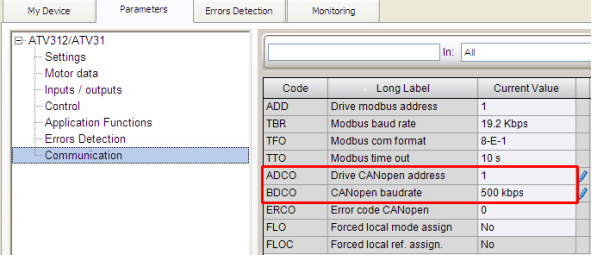
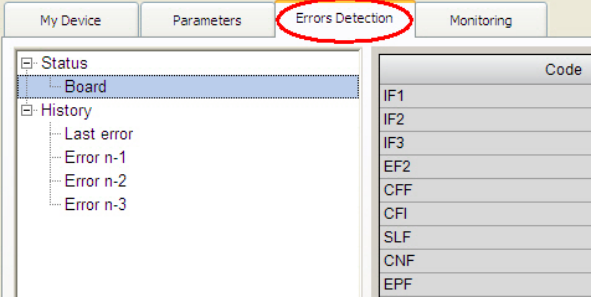
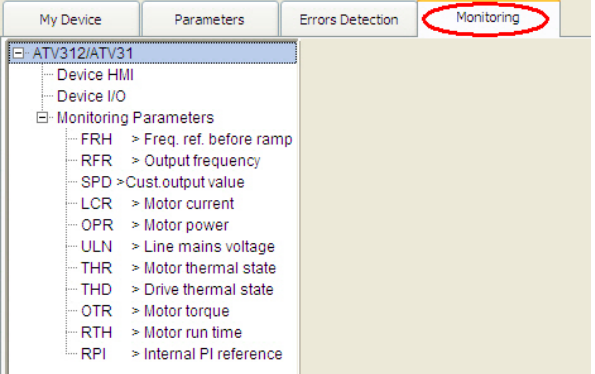

<p>11</p>	<p>The cable TCSNCSNAM3M00P is a USB cable and is linked by Windows to a virtual COM port. To identify the right COM port you need to open the <b>System Properties</b> dialog by clicking on <b>Windows key + Pause</b> button.</p> <p>Open up the <b>Hardware</b> tab and click on <b>Device Manager</b>.</p>	
<p>12</p>	<p>In the <b>Device Manager</b> click on <b>Ports (COM &amp; LPT)</b>.</p> <p>The cable is listed with the alias name <b>TSX C USB 485</b> with the corresponding port in brackets. In this case the port <b>COM15</b> is assigned to it.</p>	
<p>13</p>	<p>You can perform a test of the connection by clicking on <b>Test</b>.</p>	

<p><b>14</b></p>	<p>If the connection was successfully tested you will get the confirmation inside the screen with a green hook.</p> <p>Close this dialog box by clicking <b>OK</b>.</p>	
<p><b>15</b></p>	<p>Select:</p> <p><b>Communication-&gt;Connect to Device</b></p> <p>to go to the connection dialog.</p>	
<p><b>16</b></p>	<p>If the configuration of the device is not same as given in the file. Then user gets the following warning. Click <b>Yes</b>.</p>	

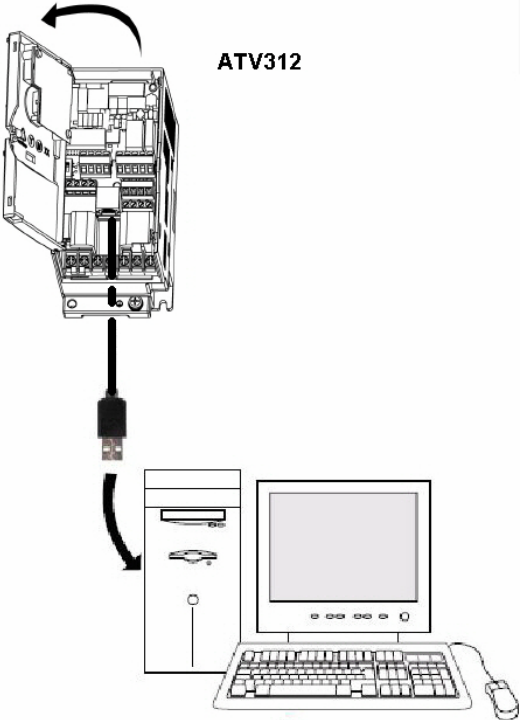
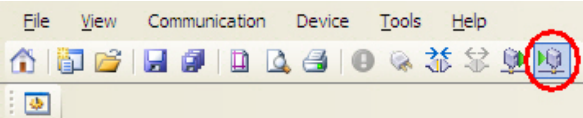



17	Press <b>Alt + F</b> to continue.	
18	The Main-Window of SoMove consists of four tabs. The tab, <b>MyDevice</b> , includes the basic information about the device.	
19	The tab, <b>Parameters</b> , shows a table of all adjustable parameters.	
20	<p>In the <b>Motor data</b> area enter the values from the motor name plate of your connected motor.</p> <p><b>Note:</b> These parameters here are for the machine described in this example only. In all likelihood, you will need to adapt these parameters for your specific machine.</p>	



<p><b>21</b></p>	<p>To adapt the CANopen communication settings go to the entry <b>Communication</b>.</p> <p>In this example we use CANopen address <b>ADCO: 1</b> CANopen baudrate <b>BDCO: 500 kbps</b></p> <p><b>Note:</b> The other Altivars will get 2, 3 and 4 as CANopen address.</p>	 <table border="1"> <thead> <tr> <th>Code</th> <th>Long Label</th> <th>Current Value</th> </tr> </thead> <tbody> <tr> <td>ADD</td> <td>Drive modbus address</td> <td>1</td> </tr> <tr> <td>TBR</td> <td>Modbus baud rate</td> <td>19.2 Kbps</td> </tr> <tr> <td>TFO</td> <td>Modbus com format</td> <td>8-E-1</td> </tr> <tr> <td>TTO</td> <td>Modbus time out</td> <td>10 s</td> </tr> <tr> <td>ADCO</td> <td>Drive CANopen address</td> <td>1</td> </tr> <tr> <td>BDCO</td> <td>CANopen baudrate</td> <td>500 kbps</td> </tr> <tr> <td>ERCO</td> <td>Error code CANopen</td> <td>0</td> </tr> <tr> <td>FLO</td> <td>Forced local mode assign</td> <td>No</td> </tr> <tr> <td>FLOC</td> <td>Forced local ref. assign.</td> <td>No</td> </tr> </tbody> </table>	Code	Long Label	Current Value	ADD	Drive modbus address	1	TBR	Modbus baud rate	19.2 Kbps	TFO	Modbus com format	8-E-1	TTO	Modbus time out	10 s	ADCO	Drive CANopen address	1	BDCO	CANopen baudrate	500 kbps	ERCO	Error code CANopen	0	FLO	Forced local mode assign	No	FLOC	Forced local ref. assign.	No
Code	Long Label	Current Value																														
ADD	Drive modbus address	1																														
TBR	Modbus baud rate	19.2 Kbps																														
TFO	Modbus com format	8-E-1																														
TTO	Modbus time out	10 s																														
ADCO	Drive CANopen address	1																														
BDCO	CANopen baudrate	500 kbps																														
ERCO	Error code CANopen	0																														
FLO	Forced local mode assign	No																														
FLOC	Forced local ref. assign.	No																														
<p><b>22</b></p>	<p>The tab, <b>Errors Detection</b>, shows the <b>current detected faults</b>.</p>																															
<p><b>23</b></p>	<p>The tab <b>Monitoring</b>, is used to monitor the parameter values of the drive. Click on the parameter to be observed and then click again on the plane on the right hand side.</p>																															
<p><b>24</b></p>	<p>After entering the needed parameter save it with:</p> <p><b>File-&gt; Save</b></p> <p>or use the icon in the toolbar, to save your configuration.</p>																															

## Download the Configuration

1	<p>Check that the PC is connected to the Altivar drive.</p> <p>For this use the cable set <b>TCSMCNAM3M002P</b>.</p>	 <p>ATV312</p>
2	<p>To download the configuration to the drive click on icon in the toolbar:</p> <p><b>Store to Device.</b></p>	
3	<p>A progress bar is displayed during the download.</p> <p>Click OK.</p>	
4	<p>For the ATV312 to operate with the new address or Baudrate, a power cycle (on, off, on) is required.</p>	


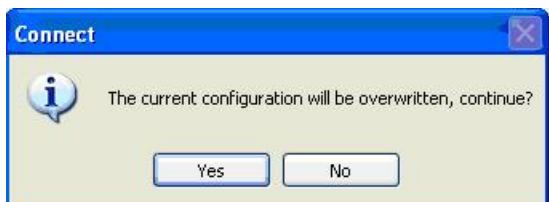

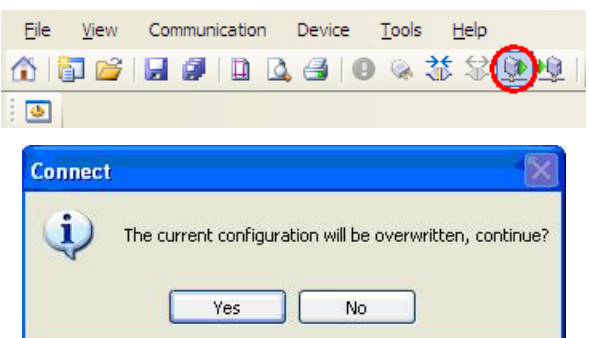
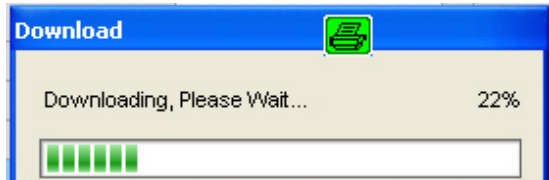
## WARNING

### UNINTENDED EQUIPMENT OPERATION

After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

**Failure to follow these instructions can cause death, serious injury or equipment damage.**

**Connect to device / upload the configuration**

1	<p>To connect to the device click the <i>Connect to Device</i> icon in the toolbar</p> <p>Or select:</p> <p><b>Communication-&gt;Connect to Device</b></p>	
2	<p>To upload the configuration to the PC answer the question that follows with</p> <p><b>Yes.</b></p> <p>The configuration in the PC file will be overwritten.</p>	
3	<p>You can also upload the configuration from the device into PC by clicking on the <i>Load Values from Device</i> icon</p>  <p>And answer the question with <b>Yes.</b></p>	
4	<p>A progress bar is displayed during the upload.</p>	

# TeSysU

## Introduction

This chapter presents the TeSysU motor components used in this system. They can be adapted according to the application (motor output, reversing or non-reversing drive).

Basically, the TeSysU motor control unit comprises of a:

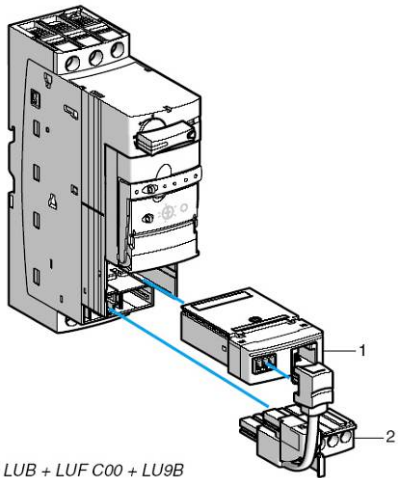
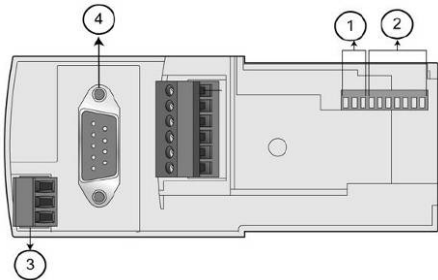
- Power base
- Control unit
- Communication module
- Coil wiring kit
- Optional: reversing block, I<sub>s</sub> limiter/isolation block and other modules

The following points should be taken into account when selecting components:

A 24 Vdc LU2B xx BL control unit **must be used**. Make sure it has the BL extension.

There are different versions of the coil wiring kit, which depend on the power base. LU9B N11C should be used if the power base has one direction of rotation (LU2Bxx) and LU9M RL should be used if the power base has two directions of rotation (LU2Bxx).

## TeSysU

1	<div>TeSysU</div> <div>Power base LU2B12BL</div> <div>Control unit LUCA05BL</div> <div>Communication module for CANopen LULC08 (1)</div> <div>Coil wiring kit LU9MRL (2)</div>	 <p>LUB + LUF C00 + LU9B</p>
2	<div>TeSysU CANopen communication module</div> <div>LULC08</div> <div>The communication module is connected to the CANopen bus using cable.</div> <div>TSXCANCADD1</div>	 <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> </div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> </div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> </div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> </div>

3

TeSysU CANOpen communication module

LULC08

The baud rate is set to **500 kbps**.

SW10	SW9	SW8	Baud Rate
0	0	0	10 kbps
0	0	1	20 kbps
0	1	0	50 kbps
0	1	1	125 kbps
1	0	0	250 kbps (default)
1	0	1	500 kbps
1	1	0	800 kbps
1	1	1	1,000 kbps

4

The following address is used: CANOpen **Node ID 5**

SW7	SW6	SW5	SW4	SW3	SW2	SW1	Address
0	0	0	0	0	0	0	Not valid
0	0	0	0	0	0	1	1 (default)
0	0	0	0	0	1	0	2
0	0	0	0	0	1	1	3
0	0	0	0	1	0	0	4
0	0	0	0	1	0	1	5

5

Note: TeSysU needs 24 Vdc on CANopen cable to operate. See the chapter:

Communication: CANopen TAP: TSXCANTDM4 wiring.

## Appendix

### Detailed Component List

The following tables are a list of the main components of the Optimized CANopen Twido architecture.

#### Hardware components – Group 1: Cabinet (Circuitry, 24 Volt Supply, etc.)

	Pos.	Qty.	Description	Part Number	Rev./ Vers.
Cabinet	1.1	1	Cabinet with mounting plate, 800 x 1200 x 400 mm (W x H x D)	NSYS3D12840P	
	1.2	1	Filter fan, 230 Vac, 165 m³/h	NSYCVF165M230PF	
	1.3	2	Outlet grille	NSYCAG223LPF	
	1.4	1	Thermostat for Fan, 0..60°C, 1 NO	NSYCCOTHO	
Optional	1.5	1	Phaseo Power supply 230 Vac / 24 Vdc, 5 A	ABL8RPS24050	
	1.6	1	Phaseo Power supply 230 Vac / 24Vdc, 5 A	ABL4RSM24050	

#### Hardware component – Group 2: Main Switch

	Pos.	Qty.	Description	Part Number	Rev./ Vers.
Main switch	2.1	1	Compact main switch 3 pole 400 Vac	NSX100F	

#### Hardware components – Group 3: Control components

	Pos.	Qty.	Description	Part Number	Rev/ Vers.
Controller and I/O modules	3.1	1	Twido modular controller, 12 IN / 8 OUT	TWDLMDA20DRT	FW 5.20
	3.2	1	CANopen master interface	TWDNCO1M	
	3.3	1	TM2 digital I/O expansion module, 16 IN / 8 OUT	TM2DMM24DRT	
	3.4	1	TM2 analog I/O expansion module, 4 IN / 2 OUT	TM2AMM6HT	
	3.5	1	TM2 analog I/O expansion module for thermocouples and thermo probe	TM2ALM3LT	

#### Hardware components – Group 4: Drive and motor starter components

	Pos.	Qty.	Description	Part Number	Rev/ Vers.
Drives & Motor Starter	4.1	4	ATV312 variable speed drives 0.37 kW	ATV312H037N4	5.1 IE 50
	4.2	4	ATV312 CANopen daisy chain option card	VW3A31208	
	4.3	4	TeSys Motor circuit breaker 2.5 A	GV2L07	
	4.4	1	TeSysU Base unit for two directions	LU2B12BL	
	4.5	1	TeSysU Control unit	LUCA05BL	
	4.6	1	TeSysU CANopen interface	LULC08	

**HMI**

Hardware components – Group 5: HMI components				
Pos.	Qty.	Description	Part Number	Rev/ Vers.
5.1	1	HMI Magelis graphical operator display, color	HMISTU655	
5.2	1	Communication cable Twido-HMI	XBTZ9780	

**E-Stop & door guarding**

Hardware components – Group 6: Safety related components				
Pos.	Qty.	Description	Part Number	Rev/ Vers.
6.1	2	Preventa Emergency Stop safety relays	XPSAC5121	
6.2	1	Harmony Emergency Stop pushbutton	XB5AS844	
6.3	2	Preventa guard switch	XCSA502	

**Pushbutton**

Hardware components – Group 7: Pushbutton				
Pos.	Qty.	Description	Part Number	Rev/ Vers.
7.1	1	Harmony Box for 1 button	XALD01	
7.2	2	Harmony Signal lamp LED white	XB5AVB1	
7.3	1	Harmony Pushbutton with LED red	XB5AW34B5	
7.4	2	Harmony Pushbutton with LED blue	XB5AW36B5	

**Software and cable**

Software-Components – Group 8				
Pos.	Qty.	Description	Part Number	Rev/ Vers.
8.1	1	TwidoSuite Software	TWDBTFU10M	V2.20.11
8.2	1	USB programming adapter	TSXCUSB485	
8.3	1	Cable for connecting programming adapter to Twido	TSXCRJMD25	
8.4	1	Vijeo Designer	VJDSNDTG5V51M	V5.1 SP1
8.5	1	Programming cable PC-HMI	BMXXCAUSBH018	
8.6	1	SoMove Lite, available by download from <a href="http://www.schneider-electric.com">http://www.schneider-electric.com</a>	web download	V2.1.4.0
8.7	1	Altivar connection cable for drives	TCSMCNAM3M002P	

## Component Protection Classes

Positioning Protection class	Component	In Field, On site			Cabinet		
					Front		inside
		IP54	IP65	IP67	IP55	IP65	IP20
	Harmony Emergency Stop switch				X		X
	Emergency Stop switch housing		X				
	Master Switch NSX 100F						X
	Rotating Door handle for master switch				X		X
	TeSys motor protection devices and load contactors						X
	Contactors and fuses						X
	Phaseo power supply						X
	Emergency stop switches XPSAC***						X
	Twido controller and expansion modules						X
	Magelis HMISTU655 touch panel					X	X
	Altivar 312 variable speed drive						X
	CANopen Taps						X
	Switch series "Harmony" in housing		IP66				
	Filter and filter fan				IP40		
	Fan thermostat						X



# Component Features

## Components **Controller Twido**

This example uses a Twido modular controller with TwidoSuite software and programming cable set.

(TWDLMDA20DRT+ TWDBTFU10EF + TSXCUSB485 + TSXCRJMD25):

- 24 Vdc
- 12 digital inputs
- 8 digital relay outputs
- expandable up to 7 modules

2 Programming Languages:

- Ladder (LD)
- Instruction List (IL)

Pre-defined functions:

- Drum control
- Fast counters up to 5kHz
- Very fast counters up to 20 kHz
- Frequency measuring 1..20 kHz
- Reserved Memory for LIFO/FIFO-processing
- PWM-/PLS-Outputs
- External controller-Start
- PID-controller



## **CANopen master module TWDNCO1M**

Master module for Twido logic controllers with:

- Control of up to 16 Slaves (depends on number of PDOs/SDOs)
- Baudrates 125 / 250 / 500 kBit/s
- Slave watchdog with Node Guarding or Heartbeat
- Configuration tool integrated in Twidosuite
- Bus configurable through the backplane of the controller
- Integrated macros for quick installation
- Slim format (30mm width)
- Pluggable contacts for power supply
- Configuration by controller during power up



### Twido analog module TWDAMM3HT

- 4 x analog input 0..10 Vdc
- 4096 increments, 12-bit digital resolution
- 2x analog output 0..10 Vdc
- 4096 increments, 12-bit digital resolution



### Safety Module: Preventa XPSAC5121

Main technical characteristics:

For monitoring	Emergency stop
Max. category accord. EN954-1	4
No. of safety circuits	3 N/O
No. of additional circuits	-
Indicators	3 LED
Power supply	AC/DC 24 V
Synchro time between inputs	Infinite
Response time	< 40 ms
AC-15 breaking capacity	C300
DC-13 breaking capacity	24 Vdc / 1.5 A - L/R 50 ms
Minimum voltage and current	17 V / 10 mA
Dimensions (mm)	114 x 22.5 x 99
Connection	Captive screw-clamp terminals
Degree of protection	IP20 (terminals) IP40 (casing)



### Power Supply Phaseo: ABL8RPS24050

- 100..240 Vac / 24 Vdc, 5 A
- Slim design
- Parallel wiring possible
- Short circuit and power surge protected



### Magelis Display Terminal: HMISTU655

- High-definition TFT QVGA with 24 Vdc power supply
- 64k colors
- Serial port RJ45 RS485/232 multiprotocol interface
- 10/100 BaseT RJ45 Ethernet port interface
- 1 mini USB Device + 1 USB Host 2.0 interface
- Temperature range: 0..+ 50 °C
- Certificates: UL, CE, cULus, C-Tick



Magelis STU Small Panel

### Altivar 312 Variable Speed Drive: ATV312H037N4

The Altivar 312 drive is a frequency inverter for 3-phase squirrel cage asynchronous motors. The Altivar 312 is robust, compact, easy to use and conforms to EN 50190, IEC/EN 61800-2, IEC/EN 61800-3 standards UL/CSA certification and to CE marking.

Altivar 312 drives communicate on Modbus and CANopen industrial buses. These two protocols are integrated as standard.

Multiple units can be mounted side by side to save space.

Drives are available for motor ratings between 0.18 kW and 15 kW, with four types of power supply:

- 200 Vac to 240 Vac 1-phase, 0.18 kW to 2.2 kW
- 200 Vac to 240 Vac 3-phase, 0.18 kW to 15 kW
- 380 Vac to 500 Vac 3-phase, 0.37 kW to 15 kW
- 525 Vac to 600 Vac 3-phase, 0.75 kW to 15 kW



### TeSysU Motor starter

with CANopen interface

**TesysU LU2B12BL+ LUCA05BL + LULC08**



### Magnetic Circuit breaker: GV2L07 and GV2L08

- 2.5 A / 4.0 A  
Short circuit protected
- Magnetic cut off at 33.5 A
- lockable



### TeSysD Contactor LC1D09BD

- Rated current 9 A AC3
- 1x NO contact
- 1x NC contact
- Positive opening operation allows for use in functional safety circuits
- 24 Vdc control voltage incl. suppressor circuit



## Contact

---

<b>Homepage</b>	<a href="http://www.schneider-electric.com">http://www.schneider-electric.com</a>
As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.	

---